

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



TRƯƠNG XUÂN HÙNG

**NGHIÊN CỨU THUẬT TOÁN TỐI ƯU BẦY ONG
TRONG HOẠCH ĐỊNH ĐƯỜNG ĐI VÀ PHÂN CỤM ĐỘNG
CHO HỆ ROBOT**

LUẬN ÁN TIẾN SĨ CƠ KỸ THUẬT

Hà Nội – 2026

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

TRƯƠNG XUÂN HÙNG

**NGHIÊN CỨU THUẬT TOÁN TỐI ƯU BẦY ONG
TRONG HOẠCH ĐỊNH ĐƯỜNG ĐI VÀ PHÂN CỤM ĐỘNG
CHO HỆ ROBOT**

Ngành đào tạo: Cơ kỹ thuật

Mã số: 9520101

LUẬN ÁN TIẾN SĨ CƠ KỸ THUẬT

**NGƯỜI HƯỚNG DẪN KHOA HỌC:
1. PGS. TS. PHẠM MẠNH THẮNG**

Hà Nội – 2026

LỜI CAM ĐOAN

Tôi xin cam đoan luận án này là kết quả nghiên cứu của tôi, được thực hiện dưới sự hướng dẫn của PGS. TS. Phạm Mạnh Thắng. Các kết quả và số liệu trình bày trong luận án là hoàn toàn trung thực và chưa từng được công bố trong bất kỳ công trình của ai khác. Các nội dung trích dẫn từ các nghiên cứu khác của các tác giả khác mà tôi trình bày trong luận án đã được ghi rõ nguồn trong phần tài liệu tham khảo.

Nghiên cứu sinh:

Hà Nội:

Lời cảm ơn

Trước hết, tôi muốn bày tỏ sự biết ơn sâu sắc nhất đến PGS. TS Phạm Mạnh Thắng, thầy đã trực tiếp định hướng và hướng dẫn tôi trong suốt thời gian thực hiện luận án.

Tôi xin chân thành cảm ơn các thầy giáo, cô giáo Khoa Cơ học kỹ thuật và Tự động hoá, Trường đại học Công nghệ, Đại học Quốc Gia Hà Nội, đã tận tình chỉ bảo, cung cấp cho tôi những kiến thức vô cùng quý giá cũng như tạo điều kiện tốt nhất cho tôi về môi trường làm việc trong thời gian học tập, nghiên cứu tại Trường.

Đồng thời tôi gửi lời cảm ơn các Lãnh đạo và đồng nghiệp tại Trung tâm Vũ trụ Việt Nam, nơi tôi đang công tác, đã tạo mọi điều kiện, bố trí thời gian tốt nhất dành cho tôi trong suốt quá trình làm nghiên cứu sinh.

Cuối cùng, tôi đặc biệt cảm ơn những người thân trong gia đình, ba, mẹ, vợ, con, cùng toàn thể bạn bè đã luôn ủng hộ, chia sẻ, động viên và khích lệ tôi trong suốt quá trình học tập và nghiên cứu./.

Nghiên cứu sinh Trương Xuân Hùng

MỤC LỤC

LỜI CAM ĐOAN.....	i
DANH MỤC TỪ VIẾT TẮT	v
DANH MỤC HÌNH VẼ	viii
DANH MỤC BẢNG BIỂU	ix
DANH MỤC THUẬT TOÁN.....	xi
MỞ ĐẦU	1
Chương 1. TỔNG QUAN	7
1.1 Giới thiệu chung	7
1.1.1 Robot di động và hệ thống đa robot.....	7
1.1.2 Hoạch định đường đi trong hệ thống đa robot	10
1.1.3 Mô hình hoá bài toán tối ưu	14
1.2 Khảo sát phương pháp hoạch định đường đi	17
1.2.1 Thuật toán cổ điển.....	18
1.2.2 Phương pháp dựa trên trí tuệ tập thể.....	21
1.3 Khảo sát và phân tích thuật toán Tối ưu bầy ong	27
1.4 Khảo sát và phân tích phương pháp phân cụm	33
1.5 Thách thức và khoảng trống nghiên cứu.....	42
1.6 Kết luận chương 1	45
Chương 2. TỐI ƯU BẦY ONG TRONG HOẠCH ĐỊNH ĐƯỜNG ĐI	47
2.1 Đặt vấn đề	47
2.2 Mô hình hoá không gian làm việc	49
2.3 Cơ chế co dãn với cặp toán tử đối ngẫu.....	50
2.4 Biểu diễn đường đi và khởi tạo phỏng đoán	52
2.5 Thiết lập hàm mục tiêu	53

2.6	Ba giai đoạn của quần thể ong	54
2.7	Kỹ thuật DeZigzagPath.....	55
2.8	Quy trình thực hiện ElasticABC	56
2.9	Kết luận chương 2.....	57
Chương 3. PHƯƠNG PHÁP PHÂN CỤM ĐỘNG		59
3.1	Đặt vấn đề	59
3.2	Mô hình hoá dữ liệu đầu vào	62
3.3	Cơ sở toán học: mô hình GMM và thuật toán EM	64
3.4	Thiết lập hàm mục tiêu cân bằng tải	69
3.5	Quy trình thực hiện DCM.....	72
3.6	Kết luận chương 3.....	73
Chương 4. THỬ NGHIỆM SỐ VÀ ĐÁNH GIÁ HOẠT ĐỘNG		74
4.1	Đặt vấn đề	74
4.2	Đánh giá quần thể bầy ong	75
4.3	Thử nghiệm và đánh giá hoạt động của thuật toán ElasticABC	80
4.4	Đánh giá hoạt động của ElasticABC với ABC truyền thống.....	107
4.5	Thử nghiệm và đánh giá hoạt động của DCM.....	110
4.6	Thảo luận về thuật toán ElasticABC và DCM.....	126
4.7	Kết luận chương 4.....	131
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....		133
DANH MỤC CÔNG TRÌNH KHOA HỌC		135
DANH MỤC KẾT QUẢ NGHIÊN CỨU KHÁC.....		137
TÀI LIỆU THAM KHẢO		139

DANH MỤC TỪ VIẾT TẮT

Từ viết tắt	Dạng đầy đủ	Diễn giải nghĩa
ABC	Artificial Bee Colony	Tối ưu bầy ong
ACO	Ant Colony Optimization	Tối ưu đàn kiến
AI	Artificial intelligence	Trí tuệ nhân tạo
APF	Artificial Potential Field	Phương pháp trường thế năng nhân tạo
BA	Bat Algorithm	Thuật toán bầy dơi
BAS	Beetle Antennae Search	Thuật toán tìm kiếm bộ cánh cứng sừng dài
CBS	Conflict-Based Search	Tìm kiếm dựa trên xung đột
CDM	Cell Decomposition Method	Phương pháp phân rã ô
CS	Cuckoo Search	Tìm kiếm cuckoo
DCM	Dynamic Clustering Method	Phương pháp phân cụm động
DFS	Depth-First Search	Thuật toán tìm kiếm theo chiều sâu
DQN	Deep Q-Networks	Mạng nơ-ron Q sâu
DWA	Dynamic Window Approach	Cửa sổ động
EM	Expectation-Maximization algorithm	Thuật toán kỳ vọng tối đa
EP	Evolutionary Programming.	Lập trình tiến hóa
FA	Firefly Algorithm	Thuật toán đom đóm
FOV	Field of View	Trường nhìn

GA	Genetic Algorithm	Thuật toán di truyền
GIS	Geographic Information System	Hệ thống thông tin địa lý
GMM	Gaussian Mixture Model	Mô hình hỗn hợp Gauss
GOA	Grasshopper Optimization Algorithm	Tối ưu đàn châu chấu
GVD	Generalized Voronoi Diagram	Biểu đồ Voronoi tổng quát
IFR	International Federation of Robotics	Liên đoàn robot quốc tế
IMU	Inertial Measurement Unit	Cảm biến đo lường quán tính
IoRT	Internet of Robotic Things	Sự kết hợp giữa hệ thống đa robot với Internet vạn vật
IoT	Internet of Things	Internet vạn vật
IQR	Interquartile range	Khoảng tứ phân vị
IWO	Invasive Weed Optimization	Tối ưu cỏ dại
LiDAR	Light Detection and Ranging	Công nghệ phát hiện ánh sáng và xác định khoảng cách
MAPF	Multi-Agent Path Finding	Tìm đường đi cho nhiều tác nhân
MPC	Model Predictive Control	Điều khiển dự báo mô hình
MRS	Multi-Robot Systems	Hệ thống đa robot
MRPP	Multi-Robot Path Planning	Hoạch định đường đi cho hệ thống đa robot
MRTA	Multi-Robot Task Allocation	Phân công nhiệm vụ cho hệ thống đa robot

PCD	Probabilistic Cell Decomposition	Phương pháp phân rã ô xác suất
PRM	Probabilistic Roadmap	Bản đồ đường đi xác suất
PSO	Particle Swarm Optimization	Tối ưu bầy đàn phân tử
RNN	Recurrent Neural Network	Mạng nơ-ron hồi quy
RRT	Rapidly Exploring Random Tree	Cây ngẫu nhiên thăm dò nhanh
SA	Simulated Annealing	Luyện kim mô phỏng
SCA	Stochastic Clustering Auction	Đấu giá phân cụm ngẫu nhiên
SI	Swarm Intelligence	Trí tuệ tập thể
SLAM	Simultaneous Localization and Mapping	Định vị và lập bản đồ đồng thời
TEB	Timed Elastic Band	Co dãn thời gian
TPS	Travelling Salesman Problem	Bài toán người bán hàng
VO	Velocity Obstacle	Thuật toán vật cản vận tốc
UAV	Unmanned Aerial Vehicle	Máy bay không người lái

DANH MỤC HÌNH VẼ

Hình 1.1: Phân loại phương pháp hoạch định đường đi.....	12
Hình 1.2: Phân loại phương pháp hoạch định đường đi (tiếp theo)	12
Hình 1.3: Đường đi khả thi và đường đi tối ưu.....	16
Hình 1.4: Phân rã ô chính xác (trái) và phân rã ô xấp xỉ (phải)	19
Hình 1.5: Nguyên lý hoạt động của phương pháp APF	21
Hình 1.6: Kiến trúc điều khiển của một hệ thống đa robot.....	44
Hình 2.1: Nguyên lý thêm điểm của toán tử Dãn	50
Hình 2.2: Nguyên lý bớt điểm của toán tử Co	51
Hình 2.3: Kỹ thuật khử gấp khúc.....	55
Hình 3.1: Liên hệ giữa DCM và MRPP	61
Hình 3.2: Mô hình với ba cụm phân phối Gauss trong không gian ba chiều	64
Hình 4.1: Bản đồ với phương án vị trí vật cản khác nhau	78
Hình 4.2: Kịch bản thử nghiệm TH #1 và TH #2	82
Hình 4.3: Kết quả đường đi tối ưu cho TH #1.....	83
Hình 4.4: Kết quả đường đi tối ưu cho TH #2.....	83
Hình 4.5: Biểu đồ hộp của chi phí cho TH #1	97
Hình 4.6: Biểu đồ hộp của chi phí cho TH #2	98
Hình 4.7: Kịch bản thử nghiệm TH #3	99
Hình 4.8: Kết quả đường đi tối ưu cho TH #3.....	100
Hình 4.9: Biểu đồ phân tán của trung bình chi phí và số lượng điểm trung gian..	102
Hình 4.10: Hiện tượng gấp khúc trong đường đi	102
Hình 4.11: Đường đi khi áp dụng kỹ thuật khử gấp khúc, TH #4	105
Hình 4.12: Đường đi khi áp dụng kỹ thuật khử gấp khúc, TH #5	106
Hình 4.13: Kết quả của ElasticABC (a) và ABC truyền thống (b).....	108
Hình 4.14: Biểu đồ tần suất dữ liệu f_{objmin} và giá trị tái lập.....	114
Hình 4.15: Phân bố điểm giao cắt theo cụm	116

DANH MỤC BẢNG BIỂU

<i>Bảng 1.1: Số liệu thống kê giai đoạn 2020 đến 2023.....</i>	<i>9</i>
<i>Bảng 1.2: Phương pháp dựa trên trí tuệ tập thể</i>	<i>22</i>
<i>Bảng 1.3: So sánh thuật toán tối ưu dựa trên trí tuệ tập thể.....</i>	<i>25</i>
<i>Bảng 3.1: Phân tích ưu điểm và nhược điểm của bộ dữ liệu</i>	<i>63</i>
<i>Bảng 3.2: Phân tích mối quan hệ giữa phối hợp và cân bằng.....</i>	<i>71</i>
<i>Bảng 4.1: Thông số của thuật toán ABC.....</i>	<i>76</i>
<i>Bảng 4.2: Giá trị chi phí của phương án thứ nhất.....</i>	<i>77</i>
<i>Bảng 4.3: Số liệu thống kê của phương án 2 và 3.....</i>	<i>79</i>
<i>Bảng 4.4: Thông số môi trường.....</i>	<i>80</i>
<i>Bảng 4.5: Thông số cho thuật toán ElasticABC.....</i>	<i>80</i>
<i>Bảng 4.6: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A1</i>	<i>84</i>
<i>Bảng 4.7: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A2</i>	<i>84</i>
<i>Bảng 4.8: Giá trị chi phí sau giai đoạn khởi tạo phỏng đoán.....</i>	<i>86</i>
<i>Bảng 4.9: Số lượng điểm trung gian sau giai đoạn khởi tạo phỏng đoán.....</i>	<i>87</i>
<i>Bảng 4.10: Giá trị chi phí sau giai đoạn tìm kiếm của ElasticABC.....</i>	<i>88</i>
<i>Bảng 4.11: Số lượng điểm trung gian sau giai đoạn tìm kiếm của ElasticABC.....</i>	<i>89</i>
<i>Bảng 4.12: Chỉ số thống kê chi phí và số lượng điểm trung gian, Bộ dữ liệu #A3 ..</i>	<i>91</i>
<i>Bảng 4.13: Chỉ số thống kê chi phí và số lượng điểm trung gian, Bộ dữ liệu #A4 ..</i>	<i>93</i>
<i>Bảng 4.14: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A5</i>	<i>99</i>
<i>Bảng 4.15: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A6</i>	<i>103</i>
<i>Bảng 4.16: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A7</i>	<i>104</i>
<i>Bảng 4.17: Thông số cho thuật toán ABC truyền thống.....</i>	<i>107</i>
<i>Bảng 4.18: Giá trị chi phí và số lượng điểm trung gian</i>	<i>109</i>
<i>Bảng 4.19: Số liệu của trường hợp mô phỏng thứ nhất</i>	<i>111</i>
<i>Bảng 4.20: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D1</i>	<i>113</i>
<i>Bảng 4.21: Phân bố robot theo cụm.....</i>	<i>115</i>
<i>Bảng 4.22: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D2 đến #D5</i>	<i>117</i>

<i>Bảng 4.23: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D6 và #D7</i>	<i>119</i>
<i>Bảng 4.24: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D8.....</i>	<i>121</i>
<i>Bảng 4.25: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D9 và #D10</i>	<i>122</i>
<i>Bảng 4.26: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D11 và #D12</i>	<i>125</i>
<i>Bảng 4.27: Đối chiếu đóng góp khoa học của cặp toán tử đối ngẫu Co và Dẫn...</i>	<i>126</i>
<i>Bảng 4.28: Đối chiếu đóng góp khoa học của kỹ thuật DeZigzagPath.....</i>	<i>127</i>
<i>Bảng 4.29: Đối chiếu ElasticABC với HABC-GA và IABC</i>	<i>128</i>
<i>Bảng 4.30: Đối chiếu DCM với phương pháp sử dụng K-means.....</i>	<i>129</i>

DANH MỤC THUẬT TOÁN

<i>Lớp Co dẫn với các toán tử thành phần</i>	<i>51</i>
<i>Khởi tạo phỏng đoán cho ElasticABC.....</i>	<i>53</i>
<i>Thuật toán hoạch định đường đi ElasticABC.....</i>	<i>56</i>
<i>Tìm ma trận điểm giao cắt.....</i>	<i>63</i>
<i>Khởi tạo mô hình hỗn hợp Gauss và thuật toán EM.....</i>	<i>68</i>
<i>Thuật toán phân cụm động DCM</i>	<i>71</i>

MỞ ĐẦU

Trước khi xuất hiện hệ robot, cách gọi ngắn gọn của hệ thống đa robot, thuật toán tìm kiếm, như Dijkstra, A*, RRT... được dùng trong hoạch định đường đi với ví dụ kinh điển là tìm lời giải cho bài toán người bán hàng (TPS). Thuật toán tìm kiếm giải pháp đường đi tối ưu trong một tập hợp nhiều đáp án đúng của bài toán. Tính xác định của môi trường là điểm đặc trưng cơ bản của loại thuật toán tìm kiếm này; những yếu tố bất định được giả thiết bỏ qua hoặc cần được xác định trước khi thực thi thuật toán. Môi trường tìm kiếm có thể được biểu diễn dưới dạng đồ thị nút hoặc đoạn thẳng, sơ đồ chia lưới hoặc phân vùng kiểu sơ đồ Voronoi, lưới tam giác Delaunay.

Vào năm 1985, Khatib đã giới thiệu phương pháp trường thế năng nhân tạo (APF), xác định hành trình di chuyển của cánh tay robot từ điểm bắt đầu tới điểm đích và có khả năng tránh được vật cản [1]. Phương pháp đề xuất một trường thế năng "ảo", trong đó robot chịu tác động đồng thời bởi lực hấp dẫn của đích đến và lực đẩy từ các vật cản. Thế năng tổng hợp tác động để hướng robot di chuyển đến đích. APF đã nhanh chóng được ứng dụng sang lớp bài toán hoạch định đường đi cho robot di động. Robot cài đặt thuật toán hoạt động với điều kiện đã xác định vị trí vật cản nhưng có thể mở rộng khả năng làm việc ngay cả khi vị trí vật cản chưa biết bằng cách trang bị thêm cảm biến phát hiện vật cản.

Khi xuất hiện hệ thống đa robot, nhiều phần tử robot hoạt động trong cùng không gian, được phân công tác vụ khác nhau, có vị trí xuất và vị trí đích đến khác nhau; sự khác biệt được "khuếch đại" cùng với quy mô của hệ thống. Khi đó, nghiên cứu về giải pháp hoạch định đường đi cần có định hướng phù hợp hơn với kiểu đối tượng như vậy. Những vấn đề cần giải quyết bao gồm:

- Tìm kiếm đường đi tối ưu cho từng phần tử, xét trong tương tác với những phần tử khác; hàm mục tiêu tối ưu về độ dài quãng đường di chuyển hoặc kết hợp với chỉ tiêu khác như thời gian, năng lượng;
- Tránh vật cản là bắt buộc, bao gồm tránh va chạm giữa các phần tử robot;

- Tính chất bất định của môi trường làm việc, tồn tại nhiều loại vật cản chưa nhận biết/đã nhận biết cũng như tĩnh/động;

- Kiến trúc kết hợp giữa tập trung và phi tập trung là sự lựa chọn phù hợp với hệ thống có quy mô lớn.

Hệ thống đa robot vận hành trong môi trường có diện tích lớn, tồn tại nhiều loại vật cản tác động đến việc hoạch định đường đi của các phần tử; đây là những đối tượng không mong muốn. Nếu dựa theo tiêu chí về vị trí thì vật cản được phân thành hai loại:

- Vật cản tĩnh: có vị trí giữ cố định trong không gian, như hòn đá, bàn, ghế...;

- Vật cản động: có khả năng thay đổi vị trí trong không gian, như con người, phương tiện khác di chuyển trong khu vực có robot đang hoạt động,...

Nếu dựa theo tiêu chí về khả năng nhận biết của robot thì vật cản được phân thành hai loại:

- Vật cản chưa nhận biết (bất định): có vị trí chưa được biết trước đó;

- Vật cản đã được nhận biết, có vị trí đã được biết đến từ trước thông qua thông tin lưu trữ về toạ độ của vật cản. Bản thân các robot đóng vai trò như những vật cản động và được nhận biết trong hoạch định đường đi.

Diện tích hoạt động lớn cũng như biến động liên tục trong môi trường vận hành đã đặt ra vấn đề đảm bảo "nhận thức" của mỗi phần tử robot. Chúng phải tự thu thập dữ liệu môi trường từ các cảm biến tích hợp trong robot nhưng đồng thời cần thêm thông tin chia sẻ từ các phần tử khác. Do đó, yêu cầu tối ưu toàn cục đối với hoạch định đường đi, theo một hàm mục tiêu cụ thể, cần được giải quyết đồng thời với việc ra quyết định ứng phó với những biến đổi cục bộ tức thời tại mỗi robot. Bên cạnh đó, khối lượng dữ liệu xử lý gia tăng nhanh chóng cùng với độ lớn của diện tích hoạt động, quy mô của hệ thống, mức độ biến đổi của môi trường trong quá trình vận hành.

Một trong những định hướng nghiên cứu mới cho những vấn đề đã đề cập ở trên là nhóm các phương pháp dựa trên trí tuệ tập thể. Trong đó, lời giải tối ưu hoặc gần tối ưu cho hệ thống nhiều đối tượng có nhiều mục tiêu dựa trên một tập hợp trí thông

minh của từng cá nhân. Trong đó, Bonabeau định nghĩa "trí tuệ tập thể" là bất kỳ nỗ lực nào nhằm thiết kế thuật toán hoặc giải pháp giải quyết vấn đề theo cách phân tán lấy cảm hứng từ hành vi tập thể của bầy côn trùng có tập tính xã hội hoặc các xã hội động vật khác [2]. Một vài phương pháp trí tuệ tập thể đã được đề xuất trong khoảng 30 năm trở lại đây như: tối ưu bầy đàn phân tử (PSO), tối ưu hóa các hàm phi tuyến liên tục [3]; tối ưu đàn kiến (ACO), mô phỏng hành vi hình thành đường đi của kiến, vào năm 1991 [4]; tối ưu bầy ong (ABC), mô phỏng các hành vi kiếm ăn của ong mật, vào năm 2005 [5]; tối ưu cỏ dại (IWO), lấy cảm hứng từ việc phát triển của cỏ dại, vào năm 2006 [6]; tối ưu bầy châu chấu (GOA) vào năm 2017 [7].

Trong bối cảnh đó, nghiên cứu sinh đã thực hiện luận án với đề tài "*Nghiên cứu thuật toán tối ưu bầy ong trong hoạch định đường đi và phân cụm động cho hệ robot*". Phạm vi nghiên cứu tập trung vào hệ thống đa robot với kiến trúc điều khiển tập trung, hoạch định đường đi và phân cụm các phần tử robot. Hệ thống đa robot được xem xét bao gồm tập hợp nhiều phần tử có cấu tạo tương tự nhau, hoạt động của chúng được quản lý, giám sát từ trạm điều khiển trung tâm. Cụ thể:

- Mỗi phần tử robot được phân công tác một tác vụ, có vị trí bắt đầu và vị trí đích đến cụ thể. Các tác vụ độc lập với nhau, tương ứng với các vị trí bắt đầu và các vị trí đích đến khác nhau.

- Hoạch định đường đi được thực thi ở trạm điều khiển trung tâm để xác định được hành trình di chuyển cho từng phần tử và truyền dữ liệu hành trình xuống từng phần tử robot.

- Phần tử robot thực thi chức năng điều khiển cục bộ như: ổn định tốc độ, bám theo quỹ đạo, tự động phát hiện và tránh vật cản...

- Trong quá trình hoạt động, các phần tử robot chia sẻ thông tin về trạng thái hoạt động, dữ liệu vận hành (vị trí, hướng di chuyển, tốc độ...) với trạm điều khiển trung tâm. Ngược lại, dữ liệu vận hành của hệ thống được trạm điều khiển trung tâm gửi xuống các phần tử để cung cấp thông tin hỗ trợ cho điều khiển cục bộ.

Nghiên cứu sinh đã tập trung vào hai mục đích nghiên cứu: (1) hoạch định đường đi dựa trên thuật toán ABC, đảm bảo được tối ưu toàn cục đồng thời duy trì năng lực tìm kiếm để ứng phó với những biến đổi cục bộ; (2) Giải pháp phân vùng để hỗ trợ quản lý hệ thống đa robot; phân chia robot thành những nhóm nhỏ, chia sẻ dữ liệu độc lập theo từng nhóm để giảm khối lượng dữ liệu vận hành của hệ thống cần trao đổi. Trong đó, mục tiêu cụ thể đồng thời cũng là đóng góp chính của luận án là:

i. Cải tiến thuật toán ABC truyền thống với cốt lõi là tối ưu cấu trúc thông qua cơ chế co dãn sử dụng cặp toán tử đối ngẫu Co và Dãn, đề xuất được thuật toán ElasticABC trong hoạch định đường đi cho hệ thống đa robot;

ii. Tích hợp chiến lược khởi tạo phỏng đoán dựa trên toán tử Dãn vào trong thuật toán ElasticABC, thay thế cho hình thức khởi tạo ngẫu nhiên trong thuật toán ABC truyền thống.

iii. Tích hợp được kỹ thuật xử lý chuyên dụng (DeZigzagPath) vào trong thuật toán ElasticABC, để xử lý hiện tượng gấp khúc cục bộ thường gặp trong hành trình được tìm kiếm, hoàn thiện chất lượng đường đi thực tế.

Sự kết hợp của điểm i và ii làm cho ElasticABC đạt được khả năng cân bằng giữa khám phá và khai thác, đáp ứng mục đích nghiên cứu thứ nhất. Các đường đi của robot thích nghi một cách có hiệu quả với độ phức tạp của môi trường thông qua việc điều chỉnh có chọn lọc số lượng điểm trung gian. Ngoài ra, cả ba điểm i, ii và iii đều là đóng góp mới của nghiên cứu về thuật toán ElasticABC.

iv. Dựa theo cách tiếp cận phân cụm mềm, sử dụng phân phối hỗn hợp Gauss để mô hình hoá sự phân bố phân tử robot theo xác suất căn cứ theo dữ liệu đầu vào là tập hợp các điểm giao cắt hành trình, đề xuất được phương pháp phân cụm động (DCM) chấp nhận sự chồng lấn linh hoạt về không gian giữa các cụm cho hệ thống đa robot.

v. Đề xuất tham số chi phí δ_k làm chỉ số đánh giá đại diện cho độ phức tạp quản lý và mức độ tiêu thụ tài nguyên của từng cụm, kết hợp với hàm mục tiêu bình phương

tối thiểu f_{obj} , hoạt động như một "bộ lọc" để lựa chọn được cấu hình phân cụm có số lượng phân tử robot từng cụm cân bằng trong DCM.

Sự kết hợp giữa điểm iv và v làm cho DCM có khả năng phân chia hệ thống đa robot thành các cụm nhỏ hơn có số lượng phân tử cân bằng, đáp ứng mục đích nghiên cứu thứ hai: giảm khối lượng dữ liệu vận hành của hệ thống cần trao đổi thông qua chia sẻ độc lập theo từng cụm đồng thời duy trì được sự cân bằng giữa các cụm. Ngoài ra, điểm v là đóng góp mới của nghiên cứu này.

vi. Đề xuất được hai giải pháp kỹ thuật xác định trạng thái chuyển động cùng với điều chỉnh hướng phát hiện vật cản cho các phân tử trong hệ thống đa robot; giải pháp đã lần lượt được cấp bằng độc quyền giải pháp hữu ích số 3615 và bằng độc quyền sáng chế số 48666.

Không kể phần mở đầu và kết luận, nội dung của luận án được tổ chức gồm 04 chương:

- Chương 1 trình bày tổng quan về những chủ đề nghiên cứu có liên quan, từ giới thiệu về robot di động như là đối tượng cơ điện tử đến hệ thống đa robot và hoạch định đường đi cùng với phân vùng. Khảo sát công trình nghiên cứu về từng chủ đề cung cấp thông tin toàn diện và tương đối đầy đủ về tình hình nghiên cứu trong lĩnh vực có liên quan, đặc biệt là hoạch định đường đi, tập trung vào nhóm phương pháp trí tuệ tập thể, và phương pháp phân cụm. Dựa trên đánh giá những hạn chế hiện hữu, nghiên cứu sinh đề xuất định hướng nghiên cứu mới để đạt được hai mục tiêu nghiên cứu đã đặt ra về hoạch định đường đi dựa trên tối ưu bầy ong và phân cụm mềm. Đây là cơ sở căn bản để nghiên cứu sinh triển khai các nội dung nghiên cứu tiếp theo, được trình bày trong Chương 2 và Chương 3.

- Chương 2 tập trung phát triển giải pháp hoạch định đường đi cho hệ thống đa robot bằng cách đề xuất biến thể ElasticABC với cơ chế tối ưu cấu trúc kết hợp khởi tạo phỏng đoán, tìm kiếm đường đi thích nghi theo điều kiện môi trường. ElasticABC khắc phục hạn chế của thuật toán tối ưu bầy ong truyền thống. Kỹ thuật xử lý hậu kỳ DeZigzagPath cũng được đề xuất tích hợp để loại bỏ hiện tượng gấp khúc cục bộ.

Các khung thuật toán cơ bản của ElasticABC được xây dựng để làm cơ sở cho giai đoạn thử nghiệm tiếp theo.

- Chương 3 tập trung vào nghiên cứu giải pháp phân cụm dựa trên mô hình hỗn hợp Gauss và thuật toán EM; đề xuất phương pháp phân cụm động (DCM) với cơ chế cân bằng tải thông qua tham số chi phí và sử dụng hàm mục tiêu toàn phương. DCM được sử dụng nhằm tìm kiếm cấu hình phân cụm cân bằng. Các khung thuật toán cơ bản của DCM đã được xây dựng để làm cơ sở cho giai đoạn thử nghiệm tiếp theo.

- Chương 4 trình bày chi tiết về kiểm chứng thuật toán ElasticABC và phương pháp DCM thông qua chương trình mô phỏng số trên máy tính và thực thi trên nhiều kịch bản đa dạng. ElasticABC thử nghiệm cho hệ robot có 10 phần tử, đánh giá hoạt động trong những cấu hình vật cản khác nhau, cũng như được so sánh với thuật toán ABC truyền thống. Tiếp theo, thử nghiệm DCM tiến hành với hệ robot có tối đa 50 phần tử, đánh giá hoạt động cũng như đặc điểm phân tán không gian thông qua những cấu hình điểm giao cắt hành trình khác nhau. Các phân tích và đánh giá về kết quả thử nghiệm đã chỉ ra được hiệu quả hoạt động và chất lượng đường đi thực tế của ElasticABC cũng như chất lượng phân cụm của DCM.

Cuối cùng, luận án đã tổng hợp Danh mục các công trình khoa học có liên quan trực tiếp đến phạm vi nghiên cứu của luận án và những kết quả nghiên cứu khác mà nghiên cứu sinh đã đạt được thuộc lĩnh vực có liên quan trong giai đoạn thực hiện luận án.

Chương 1. TỔNG QUAN

Trong những thập kỷ gần đây, sự phát triển của robot di động đã thúc đẩy sự dịch chuyển từ ứng dụng đơn lẻ sang hệ thống đa robot phức tạp (MRS - multi-robot systems, multi-robots). Chúng đã và đang trở thành xu hướng chủ đạo trong nhiều lĩnh vực như hậu cần, nông nghiệp công nghệ cao và tìm kiếm cứu nạn. Việc điều phối số lượng lớn phần tử robot di chuyển an toàn và hiệu quả trong cùng một không gian làm việc đã đặt ra thách thức lớn về mặt thuật toán và kỹ thuật; cần giải pháp hoạch định đường đi hiệu quả với hệ thống đa robot cùng với giải pháp phân vùng hoặc phân cụm khi quy mô về số lượng tăng lên, hỗ trợ cho hoạt động quản lý hệ thống. Chương này giới thiệu tìm hiểu tổng quan về chủ đề liên quan đến đối tượng nghiên cứu: hệ thống đa robot, hoạch định đường đi, phân cụm. Nội dung của chương được cấu trúc để cung cấp cái nhìn toàn diện từ khái niệm cơ bản đến phương pháp giải quyết vấn đề cụ thể, trong đó bao gồm hai phần trọng tâm. Thứ nhất là khảo sát, phân tích và so sánh các nhóm phương pháp hoạch định đường đi, từ thuật toán cổ điển cho đến phương pháp trí tuệ nhân tạo hoặc phương pháp trí tuệ tập thể/mô phỏng sinh học, đặc biệt tập trung vào phương pháp tối ưu bầy ong. Thứ hai là khảo sát và phân tích các phương pháp phân cụm "cứng" hiện có, từ các kỹ thuật phân vùng không gian kinh điển đến các thuật toán dựa trên dữ liệu. Cuối cùng, thông qua việc đánh giá thách thức và khoảng trống trong các nghiên cứu hiện có, phần cuối chương đề xuất định hướng nghiên cứu cụ thể nhằm giải quyết bài toán di chuyển cho hệ thống đa robot trong môi trường phức tạp.

1.1 Giới thiệu chung

1.1.1 Robot di động và hệ thống đa robot

Kỷ nguyên robot di động được hình thành cùng với sự xuất hiện của cặp robot di động hình con rùa Elsie và Elmer (1948-1949) cùng với robot Shakey (1966-1972). Elsie và Elmer có khả năng tự tìm đường đến nguồn sáng, tránh vật cản, và tự tìm về trạm sạc khi pin yếu; minh chứng cho những hành vi thông minh có thể được tạo ra

từ những cấu trúc đơn giản. Trong khi đó, Shakey tích hợp camera, cảm biến va chạm và kết nối với một máy tính lớn, có thể phân tích môi trường, lập kế hoạch di chuyển qua nhiều bước để thực hiện nhiệm vụ phức tạp như đẩy khối hộp. Ngày nay, robot di động được coi như một hệ thống cơ điện tử có phân tầng chặt chẽ với bốn phân hệ chính: cơ khí, truyền động, nhận thức và điều khiển.

Khung cấu trúc là thành phần chính của phân hệ cơ khí, không chỉ đóng vai trò bảo vệ các thiết bị lắp đặt bên trong mà còn quyết định trọng lượng, quán tính và hiệu suất năng lượng của toàn hệ thống. Theo khía cạnh cơ học, robot di động được xem xét như một hệ nhiều vật chịu các ràng buộc hình học và động học khắt khe trong không gian giới hạn, phân tích vị trí điểm trọng tâm, mô-men quán tính và sự phân bố tải trọng. Sự phát triển của vật liệu mới đã cung cấp thêm lựa chọn để giảm khối lượng cơ khí nhưng vẫn tăng cường được đặc tính cơ học của thành phần cơ khí. Bên cạnh đó, phân hệ truyền động gắn liền với khả năng di chuyển của robot và vấn đề tối ưu hiệu suất và điều khiển chính xác. Nghiên cứu liên quan là sự kết hợp giữa mô hình toán học về động học, động lực học, phương trình ràng buộc về vị trí, vận tốc; có sự liên hệ chặt chẽ với nhận thức và điều khiển.

Hệ thống điều khiển cần các tham số được cung cấp bởi mô hình cơ học như ma trận quán tính, hệ số ma sát và các ràng buộc non-holonomic để tính toán chính xác lực hoặc mô-men xoắn cần thiết tại các bánh xe. Mô hình phản ánh chính xác đặc điểm kết cấu vật lý cho phép vòng điều chỉnh ổn định với sai số quỹ đạo thấp, đồng thời đảm bảo robot vận hành trong giới hạn bền và an toàn. Khi mở rộng quy mô sang hệ thống đa robot, mối liên hệ được phát triển thành bài toán tối ưu đa mục tiêu, việc hoạch định đường đi cân bằng được hoạt động của từng cá thể trong tổng hợp chung của cả hệ thống. Ví dụ như đường đi của từng robot là ngắn nhất và không được va chạm với vật cản.

Trong công trình của Rubio và cộng sự, robot di động được định nghĩa bởi khả năng di chuyển tự trị và có đủ sự thông minh để phản ứng và đưa ra quyết định dựa trên nhận thức về môi trường. Nhóm tác giả chỉ ra bốn nguyên tắc cơ bản của robot di động: vận động, nhận thức, tri giác và điều hướng và việc phân loại dựa theo môi

trường hoạt động: đất, nước, không khí [8]. Trong một đánh giá khác, Darmanin và Bugeja đã chỉ ra rằng nghiên cứu giai đoạn đầu tập trung vào sáu lĩnh vực ứng dụng chính: giám sát và tìm kiếm cứu nạn, di chuyển theo bầy đàn, di chuyển theo đội hình và thám hiểm, hoạt động hợp tác, nhóm có cấu tạo không đồng nhất, thi đấu đối kháng [9]. Một số công trình nghiên cứu vào cuối những năm 1980 đã đặt nền móng và góp phần hình thành khái niệm "đa robot", hệ thống đa robot bao gồm nhiều phần tử robot di động; tập trung vào kỹ thuật điều khiển, phối hợp và giao tiếp để giúp các phần tử robot phối hợp làm việc [10].

Trong các báo cáo thường niên của Liên đoàn robot quốc tế (IFR) những năm gần đây, số liệu thống kê về thị trường robot đã cho thấy mức độ phát triển nhanh và mạnh mẽ của robot di động trong năm lĩnh vực: hậu cần/kho vận, làm sạch chuyên nghiệp, y tế/chăm sóc sức khỏe, nông nghiệp. Bảng 1.1 là số liệu của các báo cáo của giai đoạn 2021 - 2024 cho các năm từ 2020 đến 2023. Xét về ngành và lĩnh vực, thị trường chính của robot dịch vụ chuyên nghiệp đang chủ yếu tập trung chủ yếu vào nơi có nhu cầu nâng cao hiệu quả và năng suất thông qua việc sử dụng robot cho các tác vụ lặp đi lặp lại, hoạt động liên tục 24/7 trong khi luôn duy trì độ tin cậy và chính xác cao.

Bảng 1.1: Số liệu thống kê giai đoạn 2020 đến 2023

Lĩnh vực chính	2020	2021	2022	2023
Logistic và vận chuyển	44k	49,5k	86k	113k
Dịch vụ khách hàng (khách sạn, nhà hàng)	15k	20k	24,5k	54,4k
Nông nghiệp	7k	7k	8k	20k
Lau dọn chuyên nghiệp	34k	12,6k	6,9k	12k
Y tế và chăm sóc sức khỏe	18k	14,8k	9,3k	6,2k

Lĩnh vực chính	2020	2021	2022	2023
Tổng số ¹	131,8k	121k	158k	205k

Một số khảo sát gần đây cho thấy sự quan tâm nghiên cứu khác về hệ thống đa robot. Hệ thống đa robot hoạt động theo mô hình trí tuệ bầy đàn, Swarm Robotics, dựa trên phương pháp điều khiển cộng tác, có cấu tạo tương đối đơn giản, hoạt động theo cách phân tán và phi tập trung [11] [12]. Một số khảo sát khác về việc sử dụng robot máy kéo dạng [13] hoặc hệ thống không đồng nhất gồm robot trên không, mặt đất và tay máy [14] trong nông nghiệp, thay thế cho máy móc truyền thống công suất lớn; đánh giá tình trạng hiện tại của MRS trong bối cảnh ứng dụng tìm kiếm và cứu nạn [15] [16]; đánh giá về hệ thống đa robot dưới nước [17]. Bên cạnh đó, kiến trúc IoRT hình thành từ tích hợp giữa MRS với Internet vạn vật (IoT), với ba lớp: vật lý, mạng và quản lý mạng, và dịch vụ ứng dụng [18].

Tóm lại, MRS là tập hợp nhiều phần tử robot được tích hợp với công nghệ tiên tiến, máy tính nhúng, cảm biến thông minh và mạng lưới truyền thông hiện đại; kiến trúc điều khiển tập trung hoặc phi tập trung. Chúng có thể chia sẻ nhiệm vụ để đạt được mục tiêu chung bằng cách trao đổi thông tin, phối hợp và hợp tác trong quá trình vận hành. Các robot có thể hoạt động trong cùng điều kiện môi trường với khả năng phân bổ không gian tốt hơn, thực hiện các nhiệm vụ phức tạp và khó khăn mà một phần tử robot đơn lẻ không thể thực hiện được. Hơn nữa, MRS có thể đạt được khả năng chịu lỗi, dự phòng và các robot có thể thực hiện các tác vụ được giao một cách đáng tin cậy, linh hoạt và có thể tiết kiệm chi phí xét về mặt tổng thể hệ thống thông qua cải thiện hiệu suất cũng như rút ngắn thời gian thực thi.

1.1.2 Hoạch định đường đi trong hệ thống đa robot

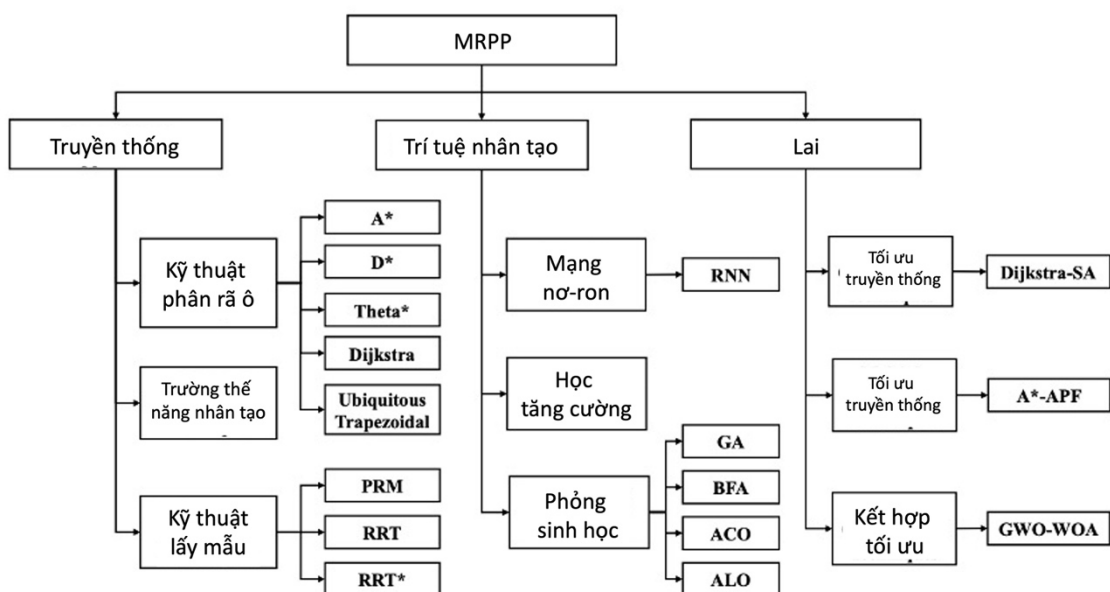
Hoạch định đường đi, hay còn gọi là lập kế hoạch đường đi, là một trong số các hướng nghiên cứu cốt lõi cho robot nói chung và robot di động cùng với hệ thống đa

¹ giá trị tổng số, bao gồm cả những lĩnh vực khác.

robot nói riêng, có lịch sử hình thành và phát triển cùng với đối tượng. Theo cách hiểu phổ biến, hoạch định đường đi là quá trình xác định hành trình di chuyển cho một robot đơn lẻ hoặc đồng thời cho nhiều phần tử robot, từ điểm xuất phát cho đến điểm đích đến cho từng robot sao cho lộ trình đảm bảo tính khả thi và được tối ưu theo tiêu chí xác định (quãng đường ngắn nhất, thời gian di chuyển, tránh va chạm...). Cho đến nay, đây vẫn là một thách thức nghiên cứu cơ bản về robot di động nói chung và hệ thống đa robot nói riêng.

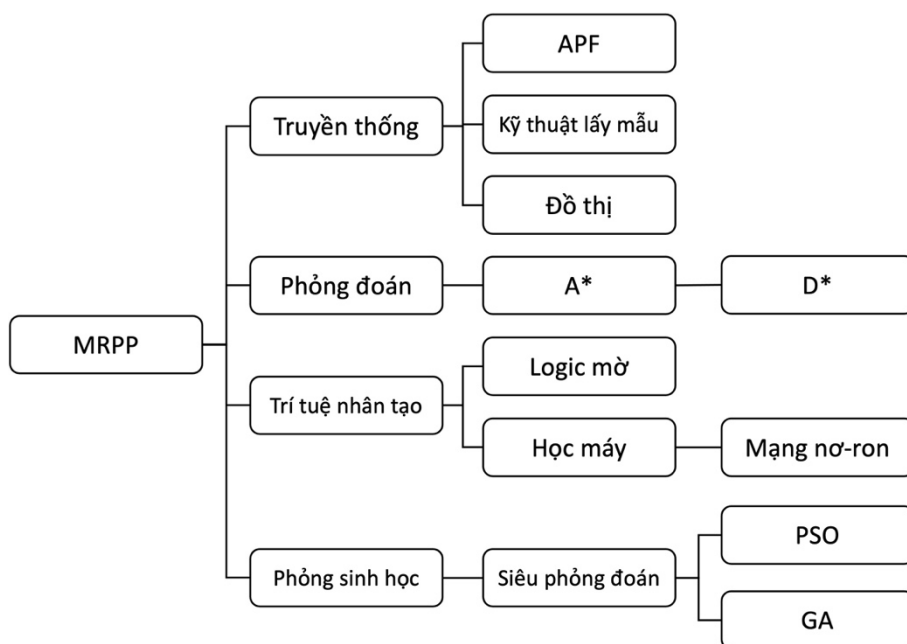
Hoạch định đường đi có vai trò rất quan trọng đối với hệ thống đa robot, trong đó phần tử robot là đồng nhất hoặc không đồng nhất, tác động trực tiếp đến khả năng phối hợp hoạt động giữa các phần tử. Trong bối cảnh này, độ phức tạp của vấn đề cần giải quyết đã tăng lên đáng kể so với trường hợp robot đơn lẻ; đường đi tối ưu được xác định không chỉ đảm bảo không va chạm với vật cản tĩnh mà còn xem xét đến chuyển động của các robot khác trong hệ thống. Hơn thế nữa, khi kích bản ứng dụng của hệ thống đa robot ngày càng phức tạp, cùng với sự gia tăng số lượng phần tử trong hệ thống hoặc xuất hiện yếu tố ngẫu nhiên trong môi trường hoạt động, giải pháp hoạch định đường đi cho hệ thống đa robot (MRPP) cổ điển đã gặp thách thức nghiêm trọng. Những phương pháp tiếp cận mới đã và đang thu hút được nhiều sự quan tâm của các nhà khoa học.

Công bố của AbuJabal và cộng sự [19] cùng với Banik và cộng sự [20] đều đánh giá một cách có hệ thống về MRPP, tập trung chủ yếu vào những nghiên cứu trong giai đoạn từ năm 2020 đến 2023. Hình 1.1 minh họa sơ đồ khối phân loại chi tiết trong công trình của AbuJabal, với ba phân nhóm chính. Thứ nhất, phương pháp xác định dựa trên thuật toán truyền thống và mô hình toán học như kỹ thuật phân rã ô (CDM) với thuật toán A* hoặc Dijkstra, APF, kỹ thuật lấy mẫu với thuật toán RRT, PRM. Thứ hai, phương pháp dựa trên học máy và những công nghệ trí tuệ nhân tạo (AI) khác, như mạng nơ-ron nhân tạo RNN, học tăng cường với kỹ thuật phổ biến Q-learning và DQN; thuật toán lấy cảm hứng từ sinh học hay phỏng đoán (heuristic), như thuật toán di truyền GA, PSO, ACO... Cuối cùng, phương pháp lai kết hợp những cách tiếp cận hoặc kỹ thuật khác nhau với những điểm mạnh riêng.



Hình 1.1: Phân loại phương pháp hoạch định đường đi

(nguồn: [19])



Hình 1.2: Phân loại phương pháp hoạch định đường đi (tiếp theo)

(nguồn: [21])

Shiwei Lin và cộng sự [21] đã đánh giá tổng quan về MRPP và các chiến lược ra quyết định, minh họa như Hình 1.2 với bốn phân nhóm chính. Thứ nhất với thứ hai lần lượt là phương pháp cổ điển như APF, RRT, hoặc kỹ thuật dựa trên đồ thị và thuật

toán tìm kiếm như A^* , D^* . Thứ ba là thuật toán lấy cảm hứng từ sinh học, như GA, PSO, ACO... được sử dụng nhiều nhất trong các nghiên cứu được đánh giá. Cuối cùng, phương pháp dựa trên AI thu hút được nhiều chú ý gần đây.

Khi MRPP được mô hình hoá bằng đồ thị vô hướng, Hang Ma [22] đã đánh giá các thuật toán tối ưu, được gọi là tìm đường đi cho nhiều tác nhân (MAPF). Chúng được phân loại thành ba nhóm chính. Thứ nhất là nhóm thuật toán rút gọn và thứ hai là nhóm thuật toán dựa trên bộ quy tắc định trước, rất hiệu quả nhưng không đảm bảo tính tối ưu và thường chỉ phù hợp cho một số loại đồ thị nhất định. Cuối cùng là thuật toán tìm kiếm như A^* , CBS (tìm kiếm dựa trên xung đột) ...

Trong khoảng 5 năm trở lại đây, cùng với ba công trình về MRPP đã nêu, ít nhất bốn công trình đánh giá tổng quan khác về hoạch định đường đi cho robot di động nói chung. Đó là công trình của Liu và cộng sự [23]; Abdulsahab và Kadhim [24]; Loganathan và Ahmad [25]; Wahab và cộng sự [26]. Xu hướng gia tăng của các nghiên cứu về phương pháp phỏng đoán từ 0% lên 95%, trong khi số lượng về phương pháp cổ điển giảm tương ứng từ 95% xuống 5% thông qua số liệu thống kê đã chỉ ra rằng trong giai đoạn 1970-2022 [24]. Các công trình cũng cho thấy xu hướng phân loại phổ biến với hai phân nhóm chính: phương pháp cổ điển và phương pháp phỏng đoán (hoặc mô phỏng sinh học), như GA, ACO, PSO, ABC. Hầu hết các nhóm nghiên cứu đều đồng thuận rằng nhóm phương pháp cổ điển bị hạn chế vì yêu cầu tính toán lớn, cần kiến thức đầy đủ về môi trường, trong khi phương pháp phỏng đoán được đánh giá là linh hoạt, mạnh mẽ hơn trong việc xử lý môi trường không chắc chắn hoặc môi trường động. Loganathan và Ahmad [25] cũng đã nhấn mạnh tính phức tạp của bài toán hoạch định đường đi, thuộc lớp NP-hard, và chỉ ra sự thống trị của phương pháp phỏng đoán trong những kịch bản khó như môi trường động và đa robot với tỷ lệ hơn 75%.

Bên cạnh đó, một cách phân loại khác dựa theo loại môi trường đã được tổng kết trong hai công trình của Liu và cộng sự [23] với Loganathan và Ahmad [25]. Đó là hoạch định toàn cục, dành cho môi trường đã biết và thực thi off-line, và hoạch định cục bộ, dành cho môi trường chưa xác định và thực thi online. Một thống kê

đáng chú ý khác cho thấy đa số các nghiên cứu được khảo sát, chiếm tỷ lệ 82%, tập trung vào môi trường tĩnh, và chỉ 18% giải quyết môi trường động [24].

Một phân nhánh khác của hoạch định đường đi là lập kế hoạch chuyển động, tạo ra các quỹ đạo thực tế tuân thủ ràng buộc động học và động lực học của robot như tốc độ hay gia tốc. Cụ thể có hai cách tiếp cận chính: phương pháp học sâu tăng cường [27] [28] [29] và phương pháp dựa theo lấy mẫu [30]. Trong đó, học sâu tăng cường được quan tâm nghiên cứu do khả năng xử lý một cách có hiệu quả sự phức tạp và ngẫu nhiên ngày càng tăng trong kịch bản ứng dụng robot hiện đại. Bên cạnh đó, hướng nghiên cứu tìm đường đi bao phủ tối ưu, tạo ra quỹ đạo không va chạm [31] [32] [33]; mục tiêu là giảm thiểu thời gian di chuyển, tốc độ xử lý, chi phí năng lượng, số lần rẽ dọc theo chiều dài đường đi, và tỷ lệ bao phủ chồng chéo thấp. Một chủ đề rất đáng chú ý khác là điều khiển theo đội hình [34].

Đóng vai trò quan trọng đối với hoạch định đường đi, khả năng nhận thức môi trường cùng với tính năng định vị và lập bản đồ đồng thời (SLAM) cũng được đánh giá là một trụ cột của robot di động [35] [36] [37] [38]. Trong đó, việc sử dụng từng cảm biến đơn lẻ tồn tại nhiều hạn chế, như lỗi tích lũy của cảm biến quán tính (IMU), hay sự nhạy cảm với môi trường của camera và bộ đo khoảng cách LiDAR; giải pháp hợp nhất dữ liệu hay hợp nhất đa cảm biến đóng vai trò quan và mang tính chất bắt buộc, kết hợp sử dụng bộ lọc như Kalman hoặc lọc hạt.

Khi mở rộng quy mô lên hệ thống đa robot, mạng truyền thông trở thành một phần không thể thiếu để đảm bảo sự phối hợp giữa các phần tử, với kiến trúc mạng, giao thức định tuyến và tích hợp IoT [18] [39]. Song song đó, trí tuệ nhân tạo trở thành công cụ hữu ích để giải quyết sự phức tạp về kỹ thuật của hệ thống đa robot, cụ thể là học tăng cường [40] và học tăng cường sâu [41] [42].

1.1.3 Mô hình hoá bài toán tối ưu

Xét trong không gian cấu hình C-space, một phần tử robot di động được quy ước là điểm duy nhất trong không gian và biểu diễn thông qua cấu hình (x, y, θ) , thể hiện vị trí và góc hướng của nó trong không gian làm việc 2D. Khi các vật cản nằm

trong không gian này được xác định và tạo thành các vùng cản (vùng cấm) thì phần không gian còn lại được gọi là vùng tự do (vùng trống). Vùng này bao gồm tất cả cấu hình đảm bảo an toàn cho phép robot di chuyển mà không bị va chạm với các vật cản. Khi đó, hoạch định đường đi được biến đổi từ xác định một vật thể có hình dạng trong không gian làm việc trở thành tìm điểm thuộc không gian cấu hình để tạo thành đường đi liên tục từ cấu hình bắt đầu $(q_{10}, q_{20}, \dots, q_{N0})$ cho đến cấu hình đích $(q_{1G}, q_{2G}, \dots, q_{NG})$ xét trong vùng không gian tự do. Hai cấu hình nói trên được khái quát khi xét cho hệ thống đa robot có N phần tử.

Xét bản chất, hoạch định đường đi là một bài toán tối ưu; xuất phát từ điều kiện ban đầu là cấu hình bắt đầu và cấu hình đích, nhiệm vụ cơ bản chỉ là tìm đường đi khả thi, tức là đến được đích mà không va chạm. Mục đích thực sự là tìm được một đường đi tối ưu dựa trên một hoặc nhiều mục tiêu. Trong đó, lựa chọn mục tiêu phụ thuộc chặt chẽ vào kịch bản hoạt động cùng với loại ứng dụng.

i. Nhóm mục tiêu cấp độ robot, liên quan đến hiệu suất cá nhân, tập trung vào chi phí và chất lượng đường đi của từng phần tử riêng rẽ. Các mục tiêu truyền thống về chi phí bao gồm:

- Quãng đường với độ dài đường đi là ngắn nhất; đây cũng là mục tiêu phổ biến nhất trong thuật toán tìm đường;

- Thời gian với thời gian di chuyển là nhanh nhất; nó không chỉ phụ thuộc vào quãng đường mà còn phụ thuộc một số yếu tố khác như vận tốc di chuyển hay mức độ phức tạp của quỹ đạo di chuyển;

- Năng lượng với mức năng lượng tiêu thụ trong một hành trình là thấp nhất; đây là một mục tiêu phức tạp vì nó phụ thuộc vào nhiều yếu tố khác như vận tốc, gia tốc, mức tải trọng; độ ma sát của bề mặt, đặc điểm địa hình.

Bên cạnh đó là một số mục tiêu về chất lượng như:

- Độ trơn của quỹ đạo di chuyển, xem xét đến những yếu tố gây hao mòn cơ khí và tổn năng lượng, hoặc làm robot thay đổi tốc độ đột ngột; đó có thể là góc rẽ hoặc đoạn cua trong một đường đi có độ dài ngắn nhất;

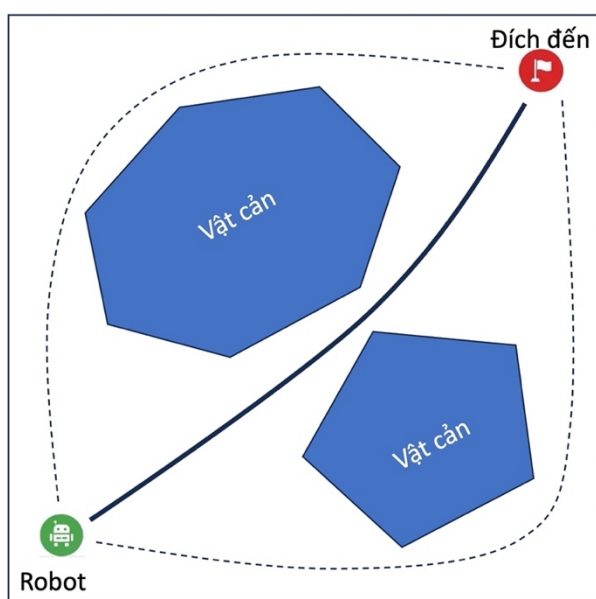
- Độ an toàn khi robot di chuyển, nhằm tối đa hóa khoảng cách tối thiểu đến các vật cản. Điều này giúp giảm thiểu rủi ro va chạm do sai số của bộ điều khiển, ngay cả khi đường đi đã khả thi về mặt lý thuyết.

ii. Nhóm mục tiêu cấp độ hệ thống, liên quan đến hiệu suất tổng thể, tập trung vào hiệu suất của cả đội thay vì các cá nhân. Đó là:

- Thời gian hoàn thành lớn nhất, là thời gian mà phần tử robot cuối cùng của hệ thống hoàn thành nhiệm vụ của mình; việc đảm bảo tham số thời gian này là tối thiểu thường được đặt ra với những hệ thống có yêu cầu giới hạn về thời gian vận hành, như trong ứng dụng kho vận và cứu hộ;

- Tổng chi phí, có thể là tổng quãng đường hoặc tổng năng lượng, để hướng đến hiệu quả hoạt động tổng thể; khi đó tổng chi phí cần đạt nhỏ nhất;

- Tổng thời gian chờ với việc cải thiện tình trạng lãng phí do chờ đợi; khi đó tổng thời gian chờ cần đạt nhỏ nhất.



Hình 1.3: Đường đi khả thi và đường đi tối ưu

Trong bài toán tối ưu đơn giản, việc lựa chọn một mục tiêu để tìm ra kết quả tối ưu duy nhất là điều thường thấy và dễ dàng đạt được. Khi đưa thêm mục tiêu thứ hai vào, vấn đề cần giải quyết trở nên phức tạp hơn. Tuy nhiên, hoạch đường đi, đặc biệt là MRPP, là một lớp bài toán khác với nhiều ràng buộc đồng thời tồn tại. Hơn thế,

những mục tiêu có ý nghĩa thực tế còn có thể có mâu thuẫn, ví dụ như đường đi ngắn nhất có thể có độ mượt kém, đường đi an toàn nhất có thể lại là đường đi dài nhất, đường đi tiết kiệm năng lượng nhất lại yêu cầu phải di chuyển chậm. Những mâu thuẫn như vậy làm khả năng tìm kiếm giải pháp tốt nhất và duy nhất đáp ứng được tất cả các mục tiêu trở nên hoàn toàn không khả thi. Điều này làm cho hoạch định đường đi trở thành bài toán tối ưu đa mục tiêu. Khi đó, giải pháp "tốt nhất" thực chất là giải pháp phù hợp nhất với yêu cầu cụ thể và đã bao hàm trong đó những đánh đổi. Hình 1.3 minh hoạ đường đi tối ưu cho một phần tử robot đơn lẻ từ tập hợp đường đi khả thi.

Công thức tổng quát của hàm mục tiêu như sau

$$J(x) = \sum_{i=1}^k \omega_i \cdot f_i(x) \quad (1.1)$$

với $J(x)$ là hàm mục tiêu tổng hợp; $f_j(x)$ là hàm mục tiêu cụ thể thứ i ; x là biến quyết định; $\omega_i \geq 0$ và thoả mãn điều kiện ràng buộc để chuẩn hoá và

$$\sum_{i=1}^k \omega_i = 1 \quad (1.2)$$

trong đó ω_i là các trọng số, đại diện cho "đánh đổi" giữa các thành phần trong hàm mục tiêu, thể hiện tầm quan trọng tương đối của mỗi thành phần xét trong tổng thể mục tiêu chung. Nhìn chung, việc lựa chọn cũng như cân bằng được mục tiêu tối ưu trong MRPP là thách thức cốt lõi và trọng tâm.

1.2 Khảo sát phương pháp hoạch định đường đi

Những kết quả khảo sát và phân tích không những phân loại được phương pháp hoạch định đường đi mà còn bộc lộ sự dịch chuyển rõ ràng trong xu hướng nghiên cứu. Cùng với sự sụt giảm đáng kể về số lượng công bố liên quan đến thuật toán cổ điển, các phương pháp trí tuệ tập thể/mô phỏng sinh học đã thu hút sự quan tâm vượt bậc; trí tuệ nhân tạo với những chủ đề học máy, học sâu, học sâu tăng cường, mạng

nơ-ron cũng rất được quan tâm. Bên cạnh đó, sự kết hợp giữa nhiều phương pháp cũng là một xu hướng không thể bỏ qua khi những phương pháp lai khai thác được ưu điểm của từng phương pháp riêng rẽ. Trong bối cảnh đó, việc khảo sát phương pháp hoạch định đường đi tập trung chủ yếu về các phương pháp mô phỏng sinh học, cụ thể là GA, PSO, ABC, ACO. Đồng thời, việc tìm hiểu một số thuật toán cổ điển, như A*, CDM, PRM, RRT, được thực hiện để mở rộng nền tảng kiến thức cơ sở phục vụ cho nghiên cứu chuyên sâu. Tiếp theo, vì đường đi toàn cục cần được đảm bảo an toàn trong môi trường thực tế, phần cuối cũng khảo sát về tránh va chạm, một trong những vấn đề kỹ thuật cơ bản nhất của hoạch định đường đi.

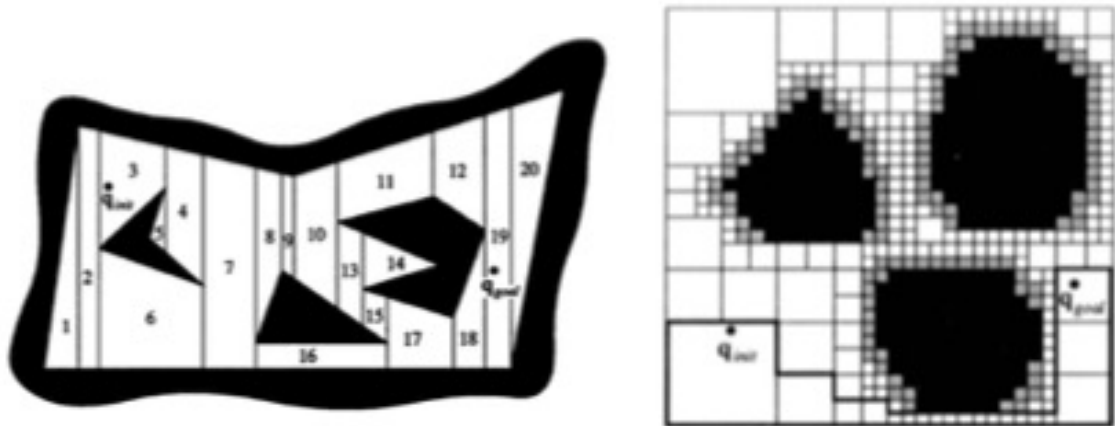
1.2.1 Thuật toán cổ điển

Thuật toán Dijkstra được công bố vào năm 1959, tìm kiếm đường đi ngắn nhất từ một điểm bắt đầu đến mọi điểm khác thuộc đồ thị với chiến lược tham lam. Thuật toán kém hiệu quả trong không gian tìm kiếm lớn, là cơ sở để Nilsson và các cộng sự công bố thuật toán A* vào năm 1968 [43], được phát triển trong dự án Shakey của Viện Nghiên cứu Stanford. Thiết kế ban đầu là tìm kiếm đường đi có tổng chi phí thấp nhất, tính từ các chi phí thành phần.

Hàm chi phí được tổng hợp từ hai thành phần:

$$f(n) = g(n) + h(n)$$

trong đó, $g(n)$ là thành phần chi phí xác định từ điểm bắt đầu đến nút đồ thị n , $h(n)$, thành phần phỏng đoán, là phần chi phí ước tính từ nút đồ thị n cho đến điểm đích. Khái niệm hàm phỏng đoán $h(n)$ có hai thuộc tính: là khả thi nếu như nó không bao giờ ước tính chi phí từ một nút đồ thị xác định đến điểm đích cao hơn chi phí thực tế nhỏ nhất; có tính nhất quán nếu như chi phí ước tính từ một nút đồ thị đến điểm đích luôn nhỏ hơn hoặc bằng chi phí di chuyển đến một nút lân cận cộng với chi phí ước tính từ đó đến điểm đích.



Hình 1.4: Phân rã ô chính xác (trái) và phân rã ô xấp xỉ (phải)

(nguồn: [44])

Xem xét trong một không gian hoạt động của robot, vị trí các vật cản trong không gian này tạo thành khu vực chiếm hữu, theo đó phần tử robot không có khả năng xâm nhập mà không gây ra va chạm. Vùng không gian tự do được hình thành từ không gian còn lại của không gian hoạt động. Latombe đã trình bày phương pháp hoạch định đường đi dựa trên nguyên tắc phân rã ô (CDM) với hai phân nhóm phân rã ô chính xác và phân rã ô xấp xỉ [44], được minh họa như Hình 1.4. Phương pháp phân rã ô xấp xỉ biểu diễn không gian tự do dưới dạng một tập hợp các ô có hình dạng đơn giản và kích thước xác định, ví dụ hình chữ nhật hoặc hình vuông. Việc chuẩn hóa hình dạng ô để đạt đồng thời hai mục tiêu: thứ nhất là sự phân chia không gian thông qua phép tính lặp đơn giản, thứ hai là ít nhạy cảm với phép tính xấp xỉ số học. Tính chất bảo toàn của không gian tự do vẫn được đảm bảo một cách cơ bản thông qua một phép xấp xỉ. Trong thực tế, lựa chọn kích thước và hình dáng của ô là vấn đề kỹ thuật cần được xác định rõ; kích thước ô lớn có thể cản trở việc tìm thấy đường đi trong một vài tình huống cụ thể. Ngược lại, kích thước nhỏ sẽ đòi hỏi thời gian tính toán tăng lên một cách có hệ thống. Gần đây, một số phương pháp phân rã ô mới đã được đề xuất, như ô hướng tâm [45], phân rã ô thích nghi [46]. Tuy nhiên, chúng vẫn được coi là một tập con của phương pháp phân rã ô xấp xỉ.

Bên cạnh CDM, biểu đồ Voronoi là kỹ thuật hình học cổ điển khác để phân rã không gian tự do thành các vùng Voronoi, mỗi vùng bao gồm tất cả các điểm gần với

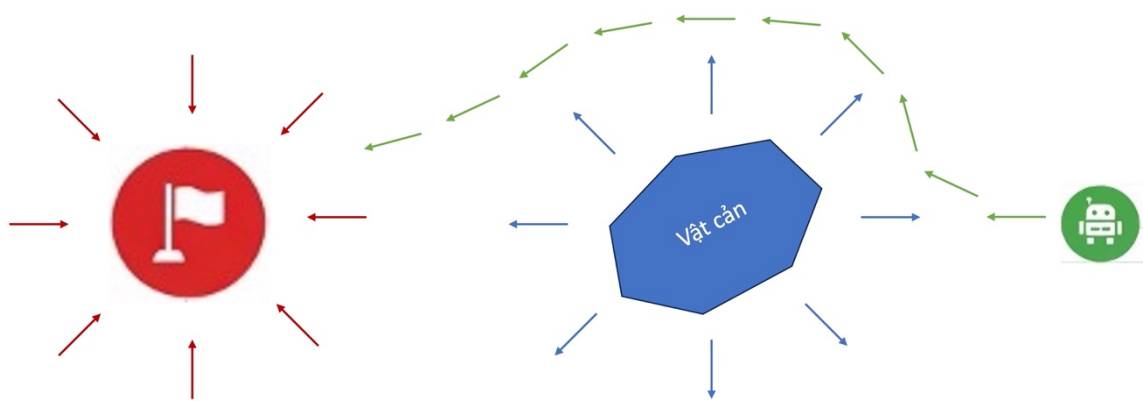
một vật cản cụ thể hơn bất kỳ vật cản nào khác. Đường tiếp giáp giữa các vùng tạo thành cấu trúc đồ thị Voronoi tổng quát (GVG). Đặc điểm quan trọng của GVG là mọi điểm trên cạnh cách đều hai hoặc nhiều vật cản, cho nên đường đi tìm được trên GVG luôn đảm bảo khoảng cách an toàn tối đa.

Khi số chiều không gian cấu hình tăng lên, tương ứng với số bậc tự do của robot, các phương pháp tìm kiếm đồ thị và phân rã không gian trở nên kém hiệu quả và yêu cầu khối lượng tính toán lớn. Đây là động lực thúc đẩy sự ra đời của phương pháp dựa trên lấy mẫu ngẫu nhiên với hai điển hình là PRM (bản đồ đường đi xác suất) [47] và RRT (cây ngẫu nhiên thăm dò nhanh), phiên bản cải tiến của PRM.

APF đã được giới thiệu lần đầu tiên vào năm 1985 trong công trình nghiên cứu của Khatib với nhiệm vụ xác định hành trình của cánh tay robot có khả năng tránh được vật cản khi chuyển động [1]. Robot hoạt động trong một trường thế năng "ảo", chịu tác động của lực hấp dẫn từ mục tiêu và lực đẩy từ các vật cản trong không gian chuyển động. Hình 1.5 minh họa cho nguyên lý lực hấp dẫn và lực đẩy tác động lên phần tử robot. Những hạn chế của APF đã được nghiên cứu cải tiến thông qua những hướng tiếp theo.

- Hướng thứ nhất thêm yếu tố mới trong thành phần lực đẩy: hệ số điều chỉnh dựa trên khoảng cách ngắn nhất giữa robot và mục tiêu trong không gian phẳng [48]; ràng buộc về hệ số khuếch đại, giá trị hằng số và được xác định bằng kinh nghiệm [49]; tất cả lực đẩy được tạo ra bởi các robot khác [50]; hệ số cho mỗi thành phần: khoảng cách từ robot đến phía bên trái của vật cản hình chữ U và đến phía bên phải của vật cản [51]. Ngoài ra, thêm giá trị ngưỡng tối thiểu trong thành phần lực hút [51].

- Hướng thứ hai với sửa đổi hàm thế năng: một hệ số ngẫu nhiên, có hướng tùy ý và giá trị ngẫu nhiên, được thêm vào lực tổng hợp [49]; trường thế năng quay [52]; hai thành phần bổ sung, là kích thích và thư giãn [53]; đề xuất hàm hoàn toàn mới [54] [55], hoặc được định nghĩa từ khoảng cách, vận tốc tương đối [56], gia tốc tương đối [57].



Hình 1.5: Nguyên lý hoạt động của phương pháp APF

- Cuối cùng, một số phương án tạo ra giải pháp mới thông qua kết hợp thêm với thuật toán khác. Ví dụ như làm phẳng đường đi bằng cách sử dụng tối ưu đường cong B-spline [49]; di chuyển theo đội hình sử dụng thuật toán bám thích nghi PID [52]; điều khiển tốc độ với hệ logic mờ FIS [51].

Hai công trình của Huang và cộng sự [58] cùng với Rafai và cộng sự [59] đã chỉ ra tránh va chạm là một yêu cầu bắt buộc xét trong bối cảnh phần tử robot cần di chuyển hiệu quả và an toàn trong môi trường có chướng ngại vật. Phương pháp APF, được xét xem là giải pháp kinh điển với nhược điểm cố hữu là khả năng bị mắc kẹt ở điểm tối ưu cục bộ; đồng thời, yếu tố "động" góp phần vào sự phức tạp của môi trường hoạt động.

1.2.2 Phương pháp dựa trên trí tuệ tập thể

Thuật toán di truyền (GA) là kỹ thuật tìm kiếm và tối ưu thuộc nhóm phương pháp tối ưu dựa trên quần thể, được phát triển chủ yếu bởi Holland từ những năm 1960 và được chính thức giới thiệu thông qua cuốn sách kinh điển "Adaptation in Natural and Artificial Systems" vào năm 1975. Dựa trên thuyết tiến hóa Darwin, GA mô phỏng lại quá trình chọn lọc tự nhiên bằng cách vận hành trên quần thể cá thể, tập hợp giải pháp tiềm năng. Chúng trải qua quá trình tiến hóa lặp đi lặp lại qua nhiều thế hệ, quần thể liên tục được cải thiện và dần hội tụ về giải pháp tốt nhất hoặc gần tốt nhất.

Bảng 1.2: Phương pháp dựa trên trí tuệ tập thể

TT	Tên	Năm
1	Tối ưu đàn kiến (ACO)	1991
2	Tối ưu bầy đàn phần tử (PSO)	1995
3	Tối ưu bầy ong (ABC)	2005
4	Tối ưu cỏ dại (IWO)	2006
5	Tối ưu bầy châu chấu (GOA)	2017

Trong khi GA dựa trên tiến hoá qua nhiều thế hệ để tìm kiếm giải pháp tối ưu, phương pháp tối ưu hóa dựa trên quần thể có một nhánh quan trọng khác, hoạt động dựa trên tương tác xã hội quần thể thế hệ hiện tại. Đó là các phương pháp dựa trên trí tuệ tập thể (SI). Trong khoảng 30 năm trở lại đây, một số phương pháp dựa trên SI đã được đề xuất, minh hoạ cụ thể như Bảng 1.2. Khái niệm này được giới thiệu trong công trình của Bonabeau và cộng sự [2], theo đó trí tuệ tập thể là bất kỳ nỗ lực nào nhằm thiết kế các thuật toán hoặc giải pháp giải quyết vấn đề theo cách phân tán lấy cảm hứng từ hành vi tập thể của các loài sinh vật có tập tính xã hội. Công trình của các tác giả đã định hình một hướng nghiên cứu hoàn toàn mới với cách tiếp cận khác từ dưới lên để đạt được các mục tiêu phức tạp.

Để áp dụng các thuật toán SI vào hoạch định đường đi, vốn là các bộ tối ưu tổng quát, cần phải "ánh xạ" bài toán vật lý sang mô hình tính toán. Quá trình này bắt đầu bằng mã hóa lời giải, tức là quy định cách một cá thể trong quần thể đại diện cho một đường đi tiềm năng. Tùy thuộc vào không gian, một lời giải có thể được mã hóa dưới dạng một véc-tơ các điểm trung gian trong không gian liên tục, hoặc một chuỗi các nút trong một đồ thị rời rạc. Trên cơ sở đó, hàm thích nghi được thiết kế để đánh giá chất lượng của lời giải đã được mã hóa, hướng dẫn quần thể tìm kiếm. Hàm phải thực hiện đồng thời hai nhiệm vụ: tối ưu hóa mục tiêu chính và xử lý ràng buộc. Ví dụ như đường đi an toàn, không xảy ra va chạm, được đánh giá dựa trên độ dài, trong khi bất kỳ đường đi nào va chạm với vật cản sẽ nhận một giá trị phạt cực lớn. Cơ chế này

đảm bảo thuật toán sẽ hội tụ về giải pháp không chỉ ngắn nhất mà còn phải an toàn tuyệt đối.

ACO được phát triển dựa trên hành vi tìm kiếm thức ăn của đàn kiến, tìm ra đường đi ngắn nhất từ tổ đến nguồn thức ăn [4]. Kiến sử dụng pheromone để đánh dấu đường đi; khi một con kiến gặp vết pheromone và đi theo thì vết pheromone được tăng cường bằng chính nó. Đường đi càng được sử dụng nhiều, mức độ hấp dẫn càng lớn. Cụ thể, khi các con kiến hoàn thành đường đi ngắn hơn nhanh hơn, vết pheromone trên con đường đó tích tụ nhanh hơn, khiến bầy đàn nhanh chóng hội tụ vào tuyến đường tối ưu. Đây là sự kết hợp giữa lựa chọn tham lam, ưu tiên các giải pháp gần, với quá trình tự xúc tác để tìm kiếm giải pháp tối ưu.

PSO được phát triển dựa trên các mô hình "sự sống nhân tạo" (A-life), đặc biệt là hành vi bầy đàn của chim và cá, dựa trên giả thuyết chia sẻ thông tin xã hội trong một quần thể mang lại lợi thế tiến hóa trong việc tìm kiếm các tài nguyên phân tán [3]. Trong quá trình các phần tử, đại diện cho những giải pháp tiềm năng, đi qua không gian nhiều chiều, vận tốc của chúng được điều chỉnh dựa trên vị trí tốt nhất mà bản thân nó đã tìm thấy (*pbest*) và vị trí tốt nhất được tìm thấy bởi bất kỳ phần tử nào trong quần thể (*gbest*). Thông qua sử dụng hai hệ số ngẫu nhiên trong phương trình vận tốc, sự cân bằng giữa khai thác giải pháp đã biết và khám phá vùng chưa biết được đảm bảo cùng với yếu tố quán tính của vận tốc.

ABC được phát triển dựa trên mô phỏng hành vi tìm kiếm mật của bầy ong. Trong mô hình này, vị trí của nguồn thức ăn đại diện cho cho một giải pháp khả thi. Bầy ong được chia thành ba nhóm: ong thợ, ong quan sát, và ong trinh sát [5]. Ong thợ được liên kết với một nguồn thức ăn cụ thể và thực hiện khai thác xung quanh giải pháp đó. Ong quan sát thông tin và chọn một nguồn thức ăn để khai thác dựa vào quy trình lựa chọn xác suất, trong đó nguồn có chất lượng cao hơn sẽ có xác suất được chọn cao hơn. Cơ chế thăm dò được kiểm soát bởi ong quan sát; nếu một nguồn thức ăn không được cải thiện sau một số lần thử thì nó sẽ bị loại bỏ và ong thợ tương ứng sẽ trở thành ong trinh sát, thực hiện tìm kiếm ngẫu nhiên một nguồn thức ăn mới.

IWO được phát triển dựa trên hành vi xâm lấn mạnh mẽ và có khả năng thích nghi cao của cỏ dại [6]. Thuật toán mô phỏng quy trình xâm lấn, bắt đầu từ một quần thể hạt giống, các giải pháp ban đầu, được phân tán ngẫu nhiên trong không gian tìm kiếm. Sau đó, mỗi cây trưởng thành sẽ tạo ra một số lượng hạt giống mới, tỷ lệ tuyến tính với hàm thích nghi của chính cây đó so với cây có độ thích nghi tốt nhất và kém nhất trong quần thể. Hạt giống mới phân tán ngẫu nhiên trong không gian xung quanh cây mẹ theo phân phối chuẩn. Độ lệch chuẩn của sự phân tán là một hàm phi tuyến và giảm dần theo thời gian, cho phép thuật toán chuyển đổi từ thăm dò toàn cục sang khai thác cục bộ. Khi quần thể đạt đến kích thước tối hạn, cơ chế loại trừ cạnh tranh giữ lại những cây có độ thích nghi cao nhất để tiếp tục sinh trưởng.

GOA được phát triển dựa trên hành vi xã hội của bầy châu chấu với hai đoạn sinh trưởng. Giai đoạn nhộng với bước đi nhỏ, chuyển động chậm, đại diện cho khai thác; giai đoạn trưởng thành với chuyển động đột ngột, tầm xa, đại diện cho thăm dò [7]. Mô hình tính toán vị trí tiếp theo của một con châu chấu dựa trên tương tác xã hội và xu hướng tiến tới mục tiêu. Trong đó, tương tác xã hội căn cứ theo vị trí của tất cả cá thể khác, tạo ra lực hút và đẩy thông qua một "vùng thoải mái". Thành phần mục tiêu hướng các cá thể về giải pháp tốt nhất ở thời điểm hiện tại. Để cân bằng giữa thăm dò và khai thác, GOA dùng hệ số thích nghi giảm dần theo số vòng lặp, giảm chuyển động tổng thể quanh mục tiêu và thu hẹp vùng tương tác xã hội; thuật toán ưu tiên thăm dò ở giai đoạn đầu và hội tụ chính xác ở giai đoạn cuối.

Nhìn chung, các phương pháp dựa trên SI được đặc trưng bởi tập hợp nguyên tắc cơ bản, đã được Bonabeau và cộng sự [2] chỉ ra lần đầu vào năm 1999. Chúng tiếp tục là nền tảng của lĩnh vực này ngay cả trong những công trình đánh giá gần đây. Khảo sát toàn diện của Nguyen [60], dựa trên 65 công bố có uy tín trong giai đoạn từ năm 2008 đến 2023, đã khẳng định sự tồn tại của các nguyên tắc cốt lõi này. Chúng bao gồm:

- Phi tập trung: hoạt động mà không cần một "bộ điều khiển" trung tâm hay thông tin tổng quan về toàn bộ hệ thống;
- Tự tổ chức: cấu trúc được hình thành qua tương tác cục bộ giữa các cá thể;

- Tự phát sinh: hành vi phức tạp, có trật tự ở cấp độ toàn cục xuất hiện từ những tương tác đơn giản ở cấp độ cá thể;

- Tương tác gián tiếp: các cá thể không giao tiếp trực tiếp với nhau mà thông qua việc biến đổi môi trường;

- Sự bền vững: những thất bại cá thể không ảnh hưởng đến hiệu suất tổng thể;

- Khả năng mở rộng cùng sự linh hoạt: thích ứng với môi trường động và duy trì hiệu suất ngay cả khi số lượng phần tử tăng lên.

Từ những phân tích đánh giá về ưu điểm, thách thức triển khai và những điểm hạn chế, Nguyen [60] đã nhận định SI là một lĩnh vực đang phát triển nhanh chóng với những kết hợp mang tính đa ngành. Bảng 1.3 trình bày so sánh theo tiêu chí: tốc độ hội tụ, khả năng mở rộng, độ phức tạp và tính phù hợp với môi trường động cho một số thuật toán SI. Ngoài những thuật toán phổ biến đã được trình bày, bảng còn mở rộng so sánh với một số phương pháp đáng chú ý khác như Thuật toán đom đóm (FA), Tìm kiếm cuckoo (CS) và Thuật toán dơi (BA) để cung cấp cái nhìn tổng quan hơn. Ba thuật toán này được đề xuất trong giai đoạn từ năm 2008 đến 2010.

Bảng 1.3: So sánh thuật toán tối ưu dựa trên trí tuệ tập thể

Thuật toán	Tốc độ hội tụ	Khả năng mở rộng	Độ phức tạp	Mức độ phù hợp với môi trường động
PSO	Nhanh ở giai đoạn đầu, có thể bị kẹt ở điểm tối ưu cục bộ.	Trung bình, hiệu suất giảm khi số chiều tăng.	Thấp, dễ cài đặt.	Trung bình, có thể phù hợp, cần thay đổi với ứng dụng thời gian thực.
ABC	Trung bình, phụ thuộc vào chiến	Cao, nhưng độ phức tạp tăng lên.	Trung bình, thách thức ở cân	Trung bình đến cao, có thể phù

Thuật toán	Tốc độ hội tụ	Khả năng mở rộng	Độ phức tạp	Mức độ phù hợp với môi trường động
	lược tìm kiếm lân cận.		bằng giữa thăm dò và khai thác.	hợp nhưng cần tinh chỉnh.
ACO	Cao, nhưng thời gian tính toán tăng khi xử lý bài toán quy mô lớn.	Trung bình, hiệu quả trong tìm kiếm tối ưu toàn cục.	Cao do cần duy trì các vết pheromone.	Cao, rất phù hợp do có bản chất phân tán.
FA	Nhanh, giỏi tránh các điểm tối ưu cục bộ.	Cao, tốt với việc triển khai song song.	Trung bình, cần tinh chỉnh tốt tham số.	Thấp đến trung bình, nhạy cảm thay đổi động, cần chiến lược phù hợp.
CS	Nhanh, hiệu quả trong cả tìm kiếm toàn cục và cục bộ.	Cao, tốt với bài toán tối ưu hóa quy mô lớn.	Trung bình đến cao, cần tinh chỉnh tốt tham số.	Thấp đến trung bình, không phù hợp nhưng có thể được điều chỉnh.
BA	Nhanh do cơ chế định vị bằng tiếng vang và điều chỉnh tần số.	Trung bình, hiệu suất suy giảm theo kích thước bài toán.	Trung bình, bao gồm các cơ chế phức tạp như điều chỉnh tần số và độ ồn.	Trung bình, có thể phù hợp nhưng cần tinh chỉnh.

1.3 Khảo sát và phân tích thuật toán Tối ưu bầy ong

Trí tuệ tập thể (SI) là những hành vi tập thể của các hệ thống phi tập trung, tự tổ chức để giải quyết các bài toán tối ưu hóa phức tạp. Hành vi thông minh toàn cục của bầy đàn không được quy định cụ thể, mà nảy sinh từ những tương tác cục bộ đơn giản giữa các cá thể và giữa các cá thể với môi trường của chúng. Rất nhiều phương pháp lấy cảm hứng từ các hệ thống tự nhiên, chẳng hạn như quá trình tạo đường đi phức tạp của đàn kiến, chuyển động đồng bộ của đàn chim và hành vi hợp tác kiếm ăn của loài ong mật. Lấy cảm hứng từ ví dụ cuối cùng này, Karaboga đã đề xuất thuật toán Tối ưu bầy ong (ABC) [61] [62]. ABC mô hình hóa một quần thể giải pháp tiềm năng dưới dạng một tập hợp các nguồn thức ăn. Để tìm ra giải pháp tốt nhất, thuật toán sử dụng một bầy ong với ba vai trò riêng biệt: ong thợ, ong quan sát và ong trinh sát, để hợp tác khám phá các nguồn thức ăn và khai thác những nguồn mang lại lợi ích cao nhất, qua đó dẫn dắt quá trình tìm kiếm hướng tới điểm tối ưu toàn cục.

Thuật toán ABC sở hữu nhiều ưu điểm nổi bật giúp nó trở thành một công cụ tối ưu hóa phổ biến. Điểm mạnh lớn nhất của ABC là tính đơn giản, dễ cài đặt và yêu cầu rất ít tham số điều khiển. Thuật toán này có khả năng tìm kiếm toàn cục rất tốt nhờ cơ chế tìm kiếm ngẫu nhiên của ong trinh sát, giúp tránh bị kẹt tại các cực tiểu địa phương trong giai đoạn đầu và có tính bền vững cao trong các môi trường phức tạp. Tuy nhiên, ABC cũng tồn tại những nhược điểm đáng kể làm hạn chế hiệu suất trong các bài toán phức tạp. Hạn chế lớn nhất là khả năng khai thác kém, dẫn đến tốc độ hội tụ chậm trong giai đoạn sau của quá trình tìm kiếm. Chiến lược tìm kiếm một chiều truyền thống của ABC thường không hiệu quả đối với các bài toán có sự phụ thuộc giữa các biến, và dễ bị lãng phí tài nguyên tính toán vào các vùng không tiềm năng nếu không có cơ chế dẫn hướng tốt.

Nhiều nghiên cứu gần đây đã tập trung cải tiến ABC để khắc phục những nhược điểm trên, đặc biệt là trong lĩnh vực lập kế hoạch đường đi cho robot. Mở đầu là định hướng kết hợp giữa ABC và Lập trình tiến hóa (EP), được khởi xướng bởi Faridi và cộng sự, để giải quyết bài toán điều hướng và lập quỹ đạo di chuyển cho nhiều robot

trong môi trường không xác định có vật cản động [63]. ABC đóng vai trò như bộ tìm kiếm cục bộ các điểm di chuyển khả thi trong phạm vi cảm biến, EP chịu trách nhiệm tối ưu hóa toàn cục để tinh chỉnh và làm trơn đường đi thông qua các toán tử đột biến như xóa, làm trơn, cập nhật và tầm nhìn. Một cơ chế phối hợp tập trung để các robot chia sẻ vị trí nhằm tránh va chạm lẫn nhau cũng được thiết lập để áp dụng cho hệ thống đa robot.

Nhận thấy công trình của Faridi chưa xem xét kỹ khoảng cách an toàn tới các vật cản khi lựa chọn điểm di chuyển, nhóm Kumar và Sikander đã phát triển phương pháp cải tiến ABC-EP, sửa đổi tiêu chí chọn nguồn thức ăn, trong đó hàm mục tiêu không chỉ dựa trên khoảng cách đến đích mà còn tính đến khoảng cách tới vật cản gần nhất để đảm bảo độ an toàn cao hơn [64]. Thuật toán cải tiến này không chỉ tạo ra đường đi an toàn hơn mà còn giảm đáng kể độ dài đường đi và chi phí thời gian tìm kiếm so với phiên bản ban đầu. Trong nghiên cứu mới nhất, nhóm Kumar và Sikander tiếp tục nâng cấp phương pháp để giải quyết các thách thức trong môi trường công nghiệp phức tạp bằng khung lai ghép ABC-PRM-EP [65]. Hai tác giả đã tích hợp phương pháp PRM vào giai đoạn đầu cùng với ABC để nhanh chóng xác định một khung đường đi cận tối ưu, sau đó mới sử dụng EP để tinh chỉnh chi tiết. Một chiến lược chờ đợi đã được thiết lập để giải quyết vấn đề phối hợp đa robot trong không gian chật hẹp; ưu tiên quyền di chuyển cho robot có tổng góc quay nhỏ hơn, qua đó giải quyết hiệu quả xung đột và va chạm trong các kịch bản công nghiệp thực tế.

Một cách kết hợp khác là với các thuật toán phỏng đoán khác nhằm khai thác ưu điểm vượt trội của từng kỹ thuật để giải quyết các hạn chế về khả năng tìm kiếm. Nghiên cứu của Li và cộng sự với thuật toán IACO-IABC, kết hợp sức mạnh tìm kiếm đường đi chi tiết của ACO với khả năng khởi tạo nhanh của ABC [66]. Phương pháp này áp dụng cơ chế tối ưu hóa đường đi loại bỏ các nút dư thừa, giảm thiểu đáng kể số lần robot phải chuyển hướng, nâng cao tính khả thi và độ mượt của quỹ đạo di chuyển. Cùng hướng tiếp cận này nhưng tập trung vào bài toán đa mục tiêu, Ye và cộng sự đã phát triển thuật toán HABC-GA bằng cách tích hợp các toán tử lai ghép

và đột biến thích ứng đặc trưng của GA vào quy trình của ABC [67]. Sự kết hợp này nhằm khắc phục hạn chế về khả năng khai thác của ABC truyền thống, giúp tăng cường khả năng tìm kiếm toàn cục và đạt được sự cân bằng tối ưu giữa ba yếu tố quan trọng: độ dài, độ an toàn và độ trơn của đường đi. Biến thể IABCBAS, kết hợp giữa thuật toán ABC và thuật toán tìm kiếm bọ cánh cứng sừng dài (BAS) [68]. Trong đó, ABC được cải tiến với ánh xạ hỗn loạn Tent trong giai đoạn khởi tạo quần thể, toán tử đột biến không đồng đều ở giai đoạn tìm kiếm với xáo trộn lớn ở giai đoạn đầu để và giảm dần xáo trộn ở giai đoạn sau để cải thiện khả năng khám phá không gian toàn cục. Thuật toán BAS được cải tiến bằng học tập ngược ngẫu nhiên và chiến lược tham lam để tinh chỉnh đường đi cục bộ. Trọng số giữa hai thuật toán được điều chỉnh nghi để cân bằng giữa tìm kiếm toàn cục và tối ưu hóa cục bộ.

Một số nghiên cứu lại tập trung nâng cao sự thông minh cho ABC bằng cách tích hợp cơ chế học tập tiên tiến để giải quyết những bài toán có độ phức tạp cao như phối hợp đa robot hay cân bằng đa mục tiêu. Cui và cộng sự đã đề xuất thuật toán ABCL, khéo léo kết hợp cơ chế của thuật toán tối ưu hóa dựa trên dạy-học (TLBO) vào giai đoạn ong quan sát [69]. Cải tiến này không chỉ tăng cường khả năng khai thác cục bộ mà còn cho phép thuật toán quy hoạch đường đi cục bộ cho nhiều robot đồng thời trong môi trường liên tục, giúp giảm thiểu hiệu quả các xung đột và va chạm giữa các robot. Bên cạnh đó, Ye và cộng sự giải quyết bài toán tối ưu hóa đa mục tiêu, cân bằng giữa độ dài ngắn nhất và độ an toàn cao nhất, thông qua thuật toán EABC [70]. Điểm đột phá của nghiên cứu là các cơ chế tối ưu hóa tự học: cơ chế tối ưu hóa dựa trên cộng tác giúp các giải pháp tốt tự hoàn thiện, trong khi cơ chế tối ưu hóa dẫn hướng bởi sự trội hướng dẫn các giải pháp kém hơn học hỏi từ các giải pháp tốt. Cách tiếp cận này giúp thuật toán duy trì sự đa dạng của quần thể và tìm ra tập hợp các giải pháp tối ưu vượt trội.

Một số nghiên cứu khác lại tập trung việc sửa đổi nền tảng tính toán và cơ chế vận hành nội tại của ABC để khắc phục hạn chế về khả năng hội tụ và xử lý dữ liệu phức tạp. Về mặt cấu trúc quần thể, Xu và cộng sự (2020) với đề xuất về khung đồng tiến hóa Co-GLABC sử dụng thông tin toàn cục để dẫn hướng [71]; áp dụng các chiến

lược tối ưu hóa riêng biệt cho hai nhóm biến độc lập và phụ thuộc. GLABC xử lý các phần độc lập về chiều, sử dụng phương trình cập nhật của thuật toán tiến hóa vi phân để tăng tốc độ hội tụ. GLABC-d xử lý các phần phụ thuộc về chiều, tích hợp chiến lược lai ghép vào giai đoạn quan sát. Trong khi đó, nghiên cứu của Zhou và cộng sự đã giới thiệu biến thể ABC-MNT để giải quyết vấn đề mất cân bằng giữa khả năng tìm kiếm và khai thác của thuật toán ABC truyền thống [72]. Bằng cách chia bầy ong thành nhiều nhóm nhỏ hơn và áp dụng các cấu trúc liên kết khác nhau như Von Neumann, Moore hay Hexagonal, thuật toán có thể điều chỉnh tốc độ lan truyền thông tin, giúp duy trì sự đa dạng của quần thể đồng thời tăng tốc độ hội tụ về lời giải tối ưu.

Thuật toán khác được đề xuất bởi Zhou và cộng sự (FLABC), tích hợp kỹ thuật phát hiện tự động đặc điểm nhẵn hoặc gồ ghề trong không gian tìm kiếm thông qua bộ mẫu dữ liệu về quần thể [73]. Ngoài ra, FLABC còn bao gồm hai cải tiến khác là chia quần thể thành nhiều nhóm con động để bao phủ các vùng tìm kiếm hiệu quả hơn và sửa đổi giai đoạn ong trình sát bằng cách sử dụng toán tử tìm kiếm lân cận toàn cầu cải tiến để bảo lưu kinh nghiệm quá khứ thay vì khởi tạo ngẫu nhiên từ đầu. Tiếp cận ở khía cạnh tham số điều khiển, Kamil và cộng sự đã giới thiệu thuật toán ADL-ABC với cơ chế giới hạn thích nghi [74]. Điểm đổi cốt lõi của thuật toán là thay thế tham số giới hạn tĩnh trong bằng cơ chế giới hạn thích ứng động, giúp thuật toán tránh những vòng lặp không cần thiết và cân bằng tốt hơn giữa khả năng tìm kiếm và khai thác. Trong khi đó, Cui và cộng sự đề xuất biến thể FOABC, tích hợp phép tính phân số vào thuật toán [75]. Tận dụng đặc điểm ghi nhớ của đạo hàm phân số, FOABC điều chỉnh giai đoạn tìm kiếm của ong quan sát, sử dụng lại thông tin từ lịch sử tìm kiếm trước đó. Một chiến lược tìm kiếm tăng cường cũng được thiết lập trong giai đoạn ong thợ để cân bằng tốt hơn giữa sự đa dạng hóa và sự tập trung trong quá trình tìm kiếm.

Biến thể MDP-ABC được đề xuất để hoạch định quỹ đạo bay trong không gian ba chiều cho UAV cánh bằng [76]; sử dụng chiến lược nhiễu loạn nhiều chiều trong giai đoạn ong thợ, chiến lược tìm kiếm vùng lân cận định hướng theo độ cong trong

giai đoạn ong quan sát. Ngoài ra chi phí đường đi được mô hình hóa độc lập theo hai hướng ngang và dọc để phù hợp với đặc tính cơ động của UAV. Một biến thể khác, tên gọi IABC, hoạch định quỹ đạo bay cho UAV trong môi trường đô thị [77]; sử dụng ánh xạ hỗn loạn Tent và nguyên lý Pareto để khởi tạo quần thể, đề xuất phương trình tìm kiếm mới và áp dụng cơ chế tiến hóa ngẫu nhiên tiếp tuyến để tránh rơi vào tối ưu cục bộ; tích hợp phương pháp nội suy spline bậc ba để tạo ra các quỹ đạo bay trơn, đáp ứng tốt các ràng buộc động học của UAV.

Cuối cùng, một số nghiên cứu tập trung tối ưu hóa thuật toán ABC cho các ràng buộc môi trường chuyên biệt và triển khai thực tế. Hai tác giả Yildirim và Akay đã giải quyết thách thức đặc thù của bài toán lập kế hoạch trên bản đồ lưới, khi đường đi tìm được thường bị gấp khúc theo các ô vuông và thiếu tính tự nhiên. Họ đề xuất thuật toán IABC tích hợp chiến lược tuyến tính hóa đường đi kết hợp với tìm kiếm cục bộ [78]. Thuật toán loại bỏ góc cua không cần thiết, giúp duỗi thẳng đoạn đường zíc-zắc thành các quỹ đạo trơn và ngắn hơn. Ở một khía cạnh khác, Promkaew và cộng sự đã đưa thuật toán ra khỏi môi trường mô phỏng để kiểm chứng trên robot tự hành (AMR) hoạt động trong môi trường trong nhà, triển khai ABC trên nền tảng hệ điều hành ROS 2 [79].

Thuật toán ABC cũng là một công cụ mạnh mẽ để giải quyết các bài toán phân bổ nhiệm vụ và lập lịch trình phức tạp trên nhiều lĩnh vực khác nhau, từ nông nghiệp thông minh đến sản xuất công nghiệp. Trong lĩnh vực nông nghiệp, nhiều biến thể của ABC được đề xuất để giải quyết những nhiệm vụ tối ưu hóa cụ thể. Chúng là MFDABC cho robot làm cỏ, hoạt động trong nhiều trang trại [80], MODABC với bài toán đa mục tiêu cho robot thu hoạch [81], CDABC cho lập lịch trình nhiều robot trong các trang trại thông minh sử dụng kỹ thuật lân cận động [82], và DABC cho việc lập lịch trình tích hợp phân công nhiệm vụ cho robot thu hoạch nông sản và xe phân phối [83]. Song song đó, ABC được áp dụng cho bài toán tối ưu hóa và lập lịch trình trong bối cảnh công nghiệp và sản xuất. Đối với các hệ thống hàn robot, một phương pháp lập lịch lại vòng kín kết hợp ABC cải tiến với học tăng cường sâu được đề xuất để xử lý hiệu quả các sự cố robot [84]. MPCCABC được ứng dụng cho dây

chuyển tháo dỡ để quản lý các kịch bản bảo trì dự phòng [85], biến thể ABC tích hợp với Q-learning để giải quyết các vấn đề lập lịch trình hai mục tiêu [86], và biến thể ABC đa mục tiêu với mô phỏng ngẫu nhiên cho hoạt động tháo dỡ có cộng tác người-robot [87]. Ngoài ra, biến thể ABC hỗ trợ quy tắc Deb tối ưu hóa bài toán đa mục tiêu của một cánh tay robot đơn lẻ xếp hàng lên pallet từ nhiều dây chuyền sản xuất [88]; một phương pháp kết hợp thuật toán Đom đóm cho phân bố toàn cục và sự lai ghép giữa thuật toán Di truyền lượng tử và ABC cho phân bố cục bộ [89]. Trong khi đó một biến thể ABC khác tích hợp các khái niệm từ phương pháp Tìm kiếm Tabu và thuật toán Miền dịch nhân tạo để giải quyết việc lập kế hoạch nhiệm vụ trong các nhà kho đa robot không đồng nhất [90].

Khả năng ứng dụng của thuật toán ABC mở rộng sang nhiều lĩnh vực kỹ thuật và khoa học khác nhau. Trong các hệ thống điều khiển, thuật toán kết hợp ABC-mờ cung cấp giải pháp cho việc điều khiển học máy thời gian thực của các tay máy robot [91]. Trong lĩnh vực hậu cần, biến thể ABC giải quyết bài toán trung tâm phân phối có dung lượng mờ trong điều kiện nhu cầu không chắc chắn [92]. Đối với các ứng dụng năng lượng, một ABC nhị phân chuyên biệt với cơ chế cập nhật đa chiều chứng tỏ hiệu quả trong việc bố trí tối ưu các tuabin gió [93]. Tương tự, CCABC giới thiệu các cơ chế tìm kiếm theo chiều ngang và chiều dọc để tăng tốc độ hội tụ, chứng tỏ hiệu quả cao trong các bài kiểm tra chuẩn và cho các nhiệm vụ phân đoạn hình ảnh [94].

Như vậy, cải tiến những yếu tố nội tại của thuật toán ABC nguyên bản là hướng nghiên cứu chủ đạo, các công bố tập trung vào "sửa chữa" nhược điểm cố hữu của thuật toán bằng cách thay đổi cấu trúc hoặc phương trình của thuật toán. Mô hình hoạt động theo ba giai đoạn (ong thợ, ong quan sát, ong trình sát) vẫn được bảo toàn. Phương trình tìm kiếm và không gian lân cận được chỉnh sửa để gia tăng tốc độ tìm kiếm hoặc tăng cường tính đa dạng để tránh mắc kẹt ở tối ưu cục bộ. Rời rạc hóa và nhị phân hóa là cải tiến nội tại khác, cho phép thuật toán ABC trở nên phù hợp hơn với những bài toán thực tế mang tính tổ hợp hoặc logic rời rạc. Bên cạnh đó, kết quả

khảo sát cho thấy xu hướng kết hợp giữa ABC với những thuật toán chuyên biệt khác để xử lý điểm yếu của thuật toán.

1.4 Khảo sát và phân tích phương pháp phân cụm

Dựa trên nguyên lý phân chia cốt lõi với ranh giới rõ ràng, các nhóm riêng biệt, không chồng lấn, mục này sẽ phân loại và khảo sát các thuật toán phân cụm cứng theo ba nhóm chính: thứ nhất là phân cụm dựa trên phân vùng không gian, thứ hai là phân cụm dựa trên logic và hành vi, và thứ ba là phân cụm dựa trên dữ liệu và các thuật toán tìm kiếm theo phỏng đoán. Trong đó, nhóm phương pháp thứ nhất tiếp cận từ góc độ hình học, với nguyên tắc cốt lõi là phân chia môi trường hoạt động thành các vùng hoặc ô rời rạc để đơn giản hóa việc quản lý. Các robot được gán vào cụm dựa trên vị trí vật lý của chúng tương ứng với các phân vùng đã được định nghĩa. CDM và biểu đồ Voronoi là những phương pháp kinh điển và phổ biến nhất cho hướng tiếp cận này.

CDM được nghiên cứu và sử dụng tương đối rộng rãi trong những nghiên cứu về hoạch định đường đi, ý tưởng cốt lõi của nó là phân chia không gian tự do, nơi robot có thể di chuyển mà không va chạm, thành tập hợp hữu hạn các ô đơn giản (vùng). Đây là bước đầu tiên để xây dựng được đồ thị liên hệ giữa các ô, làm cơ sở để thực thi thuật toán tìm đường. CDM đã được tổng hợp và giới thiệu một cách có hệ thống trong công trình của Latombe, dựa trên những nghiên cứu thuộc giai đoạn 1980-1990 [44]. Tác giả đã phân chia CDM thành hai nhóm: phân rã ô chính xác và phân rã ô xấp xỉ, được minh họa như Hình 1.4. Trong công trình đánh giá về CDM, Debnath và cộng sự [46] đã phân loại thành ba biến thể chính: phân rã ô chính xác, lưới thông thường, và phân rã ô thích nghi. Trong đó biến thể lưới thông thường có đặc điểm ô tương tự như phân rã ô xấp xỉ, các ô có kích thước cố định và không gian giống như được phủ một lưới trải đều. Phân rã ô thích nghi cho phép tạo ra những ô có kích thước không đồng đều để tăng cường hiệu quả bao phủ tại những khu vực tiếp giáp giữa không gian cản và không gian trống.

Các phương pháp phân rã ô đã phát triển đa dạng để góp phần giải quyết các bài toán lập kế hoạch đường đi và bao phủ. Một số phương pháp tập trung vào phân rã ô chính xác. Ví dụ, hình thang [95], bằng cách tạo ra các phân đoạn song song từ các đỉnh của vật cản, hình thành các ô hình thang hoặc hình tam giác; ô theo đường cày (boustrophedon) [96], bằng cách tạo ra các đường quét liên tiếp bên trong những ô đã được hình thành từ phương pháp phân rã ô chính xác. Một phương pháp hình học khác là phân rã ô hướng tâm [45], chia không gian thành các ô dạng cung tròn, đặc biệt có ích khi xử lý cho không gian chứa vật cản có cạnh thẳng đứng hoặc khi hai đỉnh có cùng tọa độ.

Một hướng tiếp cận phổ biến là phương pháp phân rã dựa trên lưới, sử dụng cấu trúc cây tứ phân (quadtree) và kỹ thuật đệ quy. Trong đó, chia nhỏ không gian cho đến khi một ô đạt được một thuộc tính duy nhất, ví dụ như ô hoàn toàn là không gian cản hoặc hoàn toàn là không gian tự do hoặc đạt đến một giới hạn phân giải là khoảng cách an toàn mong muốn của robot so với vật cản [97]. Ngoài ra, kết hợp thêm kỹ thuật gán địa chỉ trong hệ thống thông tin địa lý (GIS) được dùng để quy định địa chỉ duy nhất cho từng ô thuộc tính duy nhất [98]. Một số đề xuất chia không gian thành các ô chính kích thước xác định, ví dụ như kích thước bằng hai lần kích thước robot, sau đó mỗi ô chính lại được chia thành bốn ô con [99] [100]. Công bố [99] có thêm nút phụ mới liên kết với bốn ranh giới (bắc, nam, đông, tây) của mỗi ô chính. Công bố [100] kết hợp sử dụng biểu đồ phân vùng Voronoi dựa trên khoảng cách Manhattan trong quá trình phân rã ô. Cũng là cấu trúc cây tứ phân, thay vì tạo ra các ô có kích thước cố định, phương pháp tạo ra ô có kích thước khác nhau tương ứng với trường nhìn của robot, phụ thuộc vào độ phức tạp của địa hình [101]. Từ việc gán loại địa hình cho tất cả các ô ở lớp chi tiết nhất, sau đó gộp 4 ô con thành ô lớn hơn nếu có cùng thuộc địa hình. Nhìn chung, ưu điểm của những đề xuất dùng cây tứ phân nói trên là giảm lượng dữ liệu, tương ứng với số lượng ô, và từ đó tăng tốc quá trình tính toán sau đó.

Các phương pháp phân rã dựa trên lưới cũng có nhiều biến thể. Thay vì sử dụng lưới vuông cố định, một đề xuất lựa chọn góc phân rã tối ưu để giảm tổng diện tích

cần bao phủ; khi đó các lớp lưới song song với cạnh của khu vực bao phủ [102]. Một biến thể khác đã điều chỉnh thuật toán để gán một giá trị trọng số cho từng ô lưới, được tính toán bằng cách áp dụng thuật toán APF sửa đổi [103]. Một đề xuất khác đã thay đổi cả hình dạng ô, như sử dụng ô một phần tư quỹ đạo (QOC) thay vì ô lưới truyền thống [104].

Ngoài ra, các phương pháp kết hợp cũng một hướng tiếp cận khác. Sự kết hợp giữa phân rã hình thang, để chia không gian tự do thành các vùng, với kỹ thuật đệ quy để chia nhỏ từng vùng thành bản đồ dạng lưới [105]. Phân rã ô xác suất (PCD) kết hợp giữa phân rã ô xấp xỉ và việc lấy mẫu dựa trên xác suất, sử dụng cơ chế đánh giá "lười biếng" [106]. Theo đó, không gian cấu hình được phân rã đệ quy theo số chiều; cơ chế chỉ thị ô được lấy mẫu ngẫu nhiên để xác định trạng thái làm cơ sở phân rã, thay vì kiểm tra toàn bộ các ô hiện có. Để thay thế cấu trúc cây tứ phân, phân rã ô đa độ phân giải [107]; sử dụng biến đổi wavelet Haar, cho phép biểu diễn các vùng không gian ở các độ phân giải khác nhau đồng thời. Qua đó, cho phép phân rã chi tiết ở gần robot và phân rã thô ở các khu vực xa hơn. Cuối cùng, một biến thể phân rã ô xấp xỉ dựa trên khái niệm vùng mục tiêu (C-Target), là vùng cấu hình mà tại đó trường nhìn (FOV) của cảm biến giao với một mục tiêu [108]. Kết quả đạt được là hai loại ô chính: ô rỗng (không quan sát được) và ô quan sát (cho phép thực hiện phép đo được mục tiêu).

Biểu đồ Voronoi là một phương pháp hình học kinh điển, tính toán chia không gian đa chiều thành các vùng dựa trên mối liên hệ với một tập hợp các điểm hạt giống. Đường tiếp giáp giữa hai vùng liên tiếp được gọi là cạnh; tập hợp các cạnh tạo thành cấu trúc biểu đồ Voronoi. Các phương pháp dựa trên biểu đồ Voronoi chủ yếu thực hiện phân chia không gian để làm cơ sở cho những giai đoạn tiếp theo, như lập kế hoạch đường đi hoặc phân công nhiệm vụ. Với việc sử dụng hàm chi phí với hai yếu tố mật độ phân bố xác suất của các điểm nhiệm vụ và khoảng cách từ robot đến các điểm đó, một biến thể của biểu đồ Voronoi Tổng quát đã được đề xuất để tính toán phân vùng [109]. Sau đó, mỗi vùng Voronoi được liên kết với một robot, đảm bảo việc triển khai được thực hiện một cách cân bằng. Một biến thể khác của biểu đồ

Voronoi, 2D-VPC, tính toán bằng cách sử dụng khoảng cách Euclide tiêu chuẩn; phân chia không gian dựa trên lưới với các ô có kích thước bằng hai lần kích thước trường nhìn của cảm biến robot [110]. Phần tử robot được gán cho ô có tâm nằm gần vị trí robot nhất. Sự kết hợp giữa biểu đồ Voronoi với thuật toán phân cụm k-means cũng được đề xuất [111]. Đầu tiên, tâm cụm được xác định thông qua thuật toán k-means, dựa trên ảnh chụp vườn cây trong đó các cây được định nghĩa là nút. Biểu đồ Voronoi sử dụng tâm cụm là điểm hạt giống để phân chia không gian chứa các nút thành vùng Voronoi.

Trong khi các công trình trên chỉ tập trung giải quyết việc phân vùng cho không gian tĩnh, hướng tiếp cận khác là chiến lược phân vùng lại. Bối cảnh là việc phân vùng ban đầu có thể trở nên không hiệu quả hoặc sai lệch khi robot phát hiện vật cản mới. Phương pháp RSimExCoverage [112] sử dụng phương pháp Voronoi Manhattan để phân vùng chỉ dựa trên vị trí ban đầu của các robot ở giai đoạn đầu. Sau quá trình khám phá và bao phủ, thực hiện phân vùng lại với biểu đồ Voronoi căn cứ theo khoảng cách geodesic, đường đi ngắn nhất tránh vật cản cùng với thông tin về vật cản. Tương tự, một phương pháp phân vùng hai giai đoạn khác đã được đề xuất [113]. Cơ chế phân vùng động được bổ sung ở giai đoạn sau để xử lý vấn đề mất cân bằng do phát hiện thêm vật cản. Các robot sẽ chia sẻ nhận thức cho nhau về các chướng ngại vật đã phát hiện. Trên cơ sở đó, một thuật toán cân bằng tải dựa trên lực ảo được kích hoạt để tính toán lại các vị trí trung tâm của robot để tạo ra một biểu đồ Voronoi mới. Việc phân vùng được điều chỉnh lại cho phù hợp với các khu vực tự do đã biết, qua đó tái cân bằng khối lượng công việc.

Ngoài ra, một nghiên cứu khác đã đề xuất sử dụng biểu đồ Voronoi làm cơ chế cốt lõi để tạo ra một lộ trình tối ưu thông qua phân vùng; theo đó hai tiêu chí được cân bằng là độ dài đường đi ngắn nhất và duy trì một khoảng cách an toàn tối thiểu so với chướng ngại vật [114]. Một thuật toán được giới thiệu để xấp xỉ các vật cản bằng các điểm trên ranh giới của chúng và xây dựng Sơ đồ Voronoi từ các điểm này thay cho biểu đồ Voronoi phức tạp của các vật cản đa giác.

Thay vì dựa hoàn toàn vào ranh giới địa lý cố định, nhóm phương pháp thứ hai tập trung xây dựng cụm dựa trên các quy tắc tương tác và hành vi giữa các phần tử; các nghiên cứu có nguồn gốc và cảm hứng từ những lĩnh vực khoa học khác. Chúng có thể vận dụng lý thuyết trò chơi để định lượng sự đóng góp của từng robot hoặc mô phỏng các nguyên tắc vật lý tự tổ chức để thúc đẩy các robot tự động hình thành nhóm một cách phi tập trung.

Trong đó, lý thuyết trò chơi cung cấp một nền tảng để định lượng hóa sự tương tác và đóng góp. Phương pháp phân cụm được đề xuất dựa trên trò chơi hợp tác giữa robot và nhiệm vụ theo một hàm chi phí đánh giá hiệu suất của bất kỳ nhóm hợp tác robot và nhiệm vụ [115]. Giá trị Shapley được dùng để tính đóng góp cận biên trung bình, lượng hoá tầm quan trọng đối với robot hoặc mức độ yêu cầu đối với nhiệm vụ. Quy tắc phân cụm là nhiệm vụ đòi hỏi khắt khe nhất, giá trị Shapley thấp nhất, cho các robot có khả năng nhất, giá trị Shapley cao nhất, đồng thời cân bằng tổng giá trị Shapley của các cụm. Quy tắc trò chơi cũng được dùng trong phương pháp phân cụm cho nhóm robot lãnh đạo, theo mức độ ưu tiên của các mục tiêu và đảm bảo không bị chồng chéo dựa trên vị trí [116]. Các robot lãnh đạo được phân phối theo cụm riêng biệt, nhóm robot đi theo được phân chia cho robot lãnh đạo để tạo đội hình mong muốn.

Dựa trên nguyên tắc vật lý cơ chế tự tổ chức, một số phương pháp phi tập trung đã được đề xuất. Phương pháp phân cụm dựa trên tự phân chia thành nhiều lớp, phần tử robot trao đổi "token ảo" với các robot lân cận, mô phỏng lại quá trình lắng đọng của các chất lỏng có mật độ khác nhau [117]. Dựa trên mật độ cục bộ, những token có trọng số cao hơn tự động tụ lại ở lớp ưu tiên, tách khỏi những token có trọng số thấp. Tiếp nối ý tưởng dựa trên token ảo, một phương pháp khác đã được đề xuất, kết hợp kỹ thuật đồng thuận phân tán để tính toán mật độ cục bộ với một cơ chế cân bằng tải [118]. Thông qua phân bố mật độ trong không gian, các robot có giá trị mật độ tương đồng sẽ được tự động tụ lại để thành cụm.

Hướng tiếp cận của nhóm phương pháp thứ ba coi các robot và nhiệm vụ như những điểm dữ liệu trong không gian trạng thái đa chiều, áp dụng các kỹ thuật từ

khoa học dữ liệu hoặc tối ưu hóa để tìm ra cấu trúc nhóm hiệu quả nhất. Mục tiêu chủ đạo là tìm lời giải thông qua tối ưu hóa một hàm chi phí xác định bằng các thuật toán tìm kiếm phỏng đoán (như PSO, SA) hoặc các thuật toán phân cụm thống kê điển hình như K-means và các biến thể của nó.

Phương pháp phân cụm tương quan, một kỹ thuật học máy, đã được đề xuất cho hệ thống đa robot [119]. Nó sử dụng một hàm chi phí, nhằm mục đích nhóm các robot tương tự vào cùng một cụm và tách ra các robot không tương tự. Nghiên cứu này giải quyết bài toán phân công nhiệm vụ cho nhiều robot. Một phương pháp phân cụm khác được đề xuất dựa trên bộ lọc phân tử cho hệ thống đa robot [120]. Các phân tử lọc được nén thành K cụm bằng phương pháp sắp xếp hai chiều đệ quy theo chiều có phương sai cao nhất. Ngoài ra, phương pháp phân cụm theo mật độ (CFSFDP) được sử dụng để phân vùng các khu vực rời rạc [121]. Đầu tiên, các trung tâm cụm (tương ứng với số lượng UAV) được xác định dựa trên tính toán khoảng cách và mật độ, tổng chi phí thời gian để bay đến và quét tất cả các khu vực lân cận. Những khu vực còn lại được phân vào từng cụm căn cứ theo tổng thời gian yêu cầu thấp nhất.

Cũng không dựa hoàn toàn vào ranh giới cố định của không gian, nghiên cứu của Arslan đã đề cập đến một cơ chế trừu tượng hoá không gian cấu hình thông qua công cụ nền tảng là phương pháp phân cụm phân cấp [122] [123]. Phương pháp này tạo ra một liên kết trực tiếp giữa không gian cấu hình (liên tục) với không gian kết hợp cây cấu trúc (rời rạc), mỗi cây dữ liệu đại diện cho một cấu trúc các cụm robot lồng nhau và khác nhau, được xác định dựa trên sự gần gũi về không gian. Trên cơ sở đó, tác giả giới thiệu khung điều hướng phân cấp và thuật toán phản ứng cho hai chế độ điều hướng riêng biệt:

- Điều hướng xuyên không gian kết hợp cây cấu trúc, cho phép hệ thống thực hiện các chuyển đổi cấu trúc một cách có chủ đích.
- Điều hướng bất biến bên trong không gian cấu hình, cho phép robot duy trì một cấu trúc cụm mong muốn trong khi di chuyển không va chạm.

Thay vì tập trung vào cấu trúc hình học hay tô-pô, một hướng tiếp cận khác trong nhóm này là sử dụng các thuật toán tối ưu phỏng đoán để tìm kiếm giải pháp phân cụm và phân bổ nhiệm vụ. Thuật toán đấu giá phân cụm ngẫu nhiên (SCA) được đề xuất để phân bổ nhiệm vụ tối ưu cho hệ thống đa robot không đồng nhất [124]. Nó sử dụng tìm kiếm chuỗi Markov kết hợp với giải thuật luyện kim mô phỏng (Simulated Annealing - SA). Một phương pháp khác, CDTA, được đề xuất dựa trên giải thuật PSO [125]. Phần tử robot đại diện cho mỗi cụm được lựa chọn theo giá trị hàm thích nghi, được tính toán theo khoảng cách Euclide từ robot đến nhiệm vụ và năng lượng còn lại của robot. Giao tiếp nội bộ trong cụm được thực hiện theo cấu trúc hình sao.

Là một thuật toán học máy không giám sát rất phổ biến, K-means được sử dụng để giải quyết bài toán phân cụm dữ liệu; nó phân chia một tập hợp các điểm dữ liệu thành K nhóm riêng biệt, điểm dữ liệu trong mỗi cụm có sự tương đồng cao hơn so với các điểm ở các cụm khác. Khả năng phân cụm này đã hỗ trợ rất tốt cho tính năng tìm đường của robot, đặc biệt là những ứng dụng nhận diện đường đi bằng hình ảnh. Thuật toán K-means, với số lượng cụm tối ưu được xác định bởi phương pháp Silhouette, phân đoạn ảnh thành hai cụm: cụm thứ nhất đại diện cho cây cối và cụm thứ hai đại diện cho đường đi và bầu trời [126]. Dữ liệu đầu vào là các đặc trưng kết cấu phức tạp từ môi trường vườn cây được xác định bằng bộ lọc Gabor thay vì dựa vào phương pháp phân đoạn RGB truyền thống; đồng thời sử dụng kỹ thuật phân tích thành phần chính (PCA) để tinh gọn bộ dữ liệu đầu vào. Một phương pháp khác sử dụng ảnh chiều sâu thay thế ảnh RGB để xử lý nhanh và phân biệt đối tượng dễ dàng hơn, đồng thời kết hợp thông tin ngữ nghĩa từ YOLOv3 với dữ liệu chiều sâu để tự động tính toán số lượng cụm phù hợp [127]. Một phương pháp lai PSO-K, kết hợp giữa PSO và thuật toán K-means, sử dụng khả năng tìm kiếm toàn cục của PSO để xác định các tâm cụm tối ưu trước khi chạy thuật toán K-means [128]. Dữ liệu đầu vào là ảnh trong không gian màu YCbCr thay vì trên không gian RGB, giúp giảm ảnh hưởng của điều kiện ánh sáng và bóng râm trên loại ảnh RGB có cường độ kênh xanh lá lớn.

Thuật toán K-means được ứng dụng rộng rãi trong bài toán phân công nhiệm vụ cho hệ thống đa robot (MRTA) với chiến lược chia nhỏ, nhằm giảm độ phức tạp tính toán và không gian trạng thái. Bài toán người du lịch đa năng (MTSP) phức tạp với N điểm nhiệm vụ thành K bài toán người du lịch (TSP) độc lập và đơn giản hơn. Trong đó, hàm mục tiêu được lập dựa trên khoảng cách giữa các nhiệm vụ; nhiệm vụ có khoảng cách càng nhỏ thì độ tương đồng càng cao và càng có khả năng được phân vào cùng một cụm [129]. Một số lựa chọn hàm mục tiêu khác như: dựa trên tổng khoảng cách bình phương giữa các nhiệm vụ và tâm cụm tương ứng, tập hợp tâm cụm ban đầu được lựa chọn một cách ngẫu nhiên [130]; hàm mục tiêu dựa trên tổng khoảng cách của mỗi cụm được cân bằng bên cạnh mục tiêu tối thiểu hóa khoảng cách so với điểm tâm bên trong mỗi cụm [131]. Trong một bài toán cụ thể về phân công nhiệm vụ cân bằng cho hệ thống đa robot (BMRTA), nhóm tác giả đã phân tích và so sánh hiệu suất của ba kỹ thuật phân cụm cụ thể: K-means, GMM, và phân cụm phân cấp [132]. Trong khi GMM gán nhiệm vụ vào cụm theo hàm phân phối xác suất thì thuật toán K-means thực hiện phân chia dữ liệu thành K cụm loại trừ lẫn nhau bằng cách gán các nhiệm vụ cho tâm cụm gần nhất. Cuối cùng, phân cụm phân cấp xây dựng một cây phân cấp và sau đó cắt cây này ở mức độ phù hợp để tạo ra số lượng cụm mong muốn. Trong kết quả thử nghiệm với bài toán cụ thể, K-means là phương pháp phù hợp nhất và có khả năng mở rộng tốt để xử lý số lượng lớn nhiệm vụ và robot trong các cấu trúc liên kết phức tạp.

Bên cạnh đó, thuật toán K-means được ứng dụng trong phương pháp phân cụm động cho những bài toán yêu cầu tính thích ứng cao như khám phá và bao phủ môi trường; đề xuất phân vùng các khu vực chưa khám phá, phân công nhiệm vụ bao phủ khu vực cho một hệ thống đa robot [133]. Số lượng cụm được tạo ra bằng với số lượng robot có sẵn, tâm cụm là vị trí đích của từng robot. Khi robot di chuyển và khu vực chưa được bao phủ thay đổi, quá trình phân vùng này mang tính chất động cho phép điều chỉnh lại vị trí điểm đích (tâm cụm). Một cách tiếp cận khác là phân chia tập hợp các vùng không gian chưa biết thành K vùng rời rạc, bằng đúng số lượng robot có sẵn; phân công nhiệm vụ khám phá cho hệ thống đa robot [134]. Hàm mục

tiêu là tối thiểu hóa tổng bình phương trong cụm, đảm bảo các ô trong cùng một vùng được nhóm lại với nhau một cách logic. Quá trình phân vùng cho phép không gian chưa biết còn lại sẽ được phân vùng lại một cách tự động, còn gọi là phân vùng động, giúp cân bằng khối lượng công việc chung giữa các robot.

Thuật toán K-means còn được cải tiến hoặc kết hợp với những thuật toán khác để đạt được yêu cầu về hiệu suất. Ví dụ như, MRTA dựa trên kết hợp giữa thuật toán K-means++ cải tiến với thuật toán PSO. Đầu tiên, K-means++ tập trung giải quyết điểm hạn chế khi chọn tâm cụm ban đầu ngẫu nhiên, bổ sung một điều kiện giới hạn về số lượng tối đa các nhiệm vụ trong mỗi cụm để giúp kết quả phân cụm ổn định hơn [135]. Tiếp theo, thuật toán PSO gán từng robot vào các cụm và sắp xếp thứ tự thực hiện các nhiệm vụ bên trong cụm một cách tối ưu. Một phương pháp lai khác dựa trên thuật toán K-means, được gọi là phương pháp tìm kiếm tham lam phân cụm sơ bộ [136]. Thuật toán Lloyd được kết hợp để xác định tập hợp các tâm cụm dựa trên hàm mục tiêu tổng bình phương trong cụm. Sau khi phân chia thành các cụm nhỏ hơn, thuật toán lập kế hoạch đường đi bên trong mỗi cụm được thực hiện để xác định hành trình ngắn nhất. Cuối cùng, một phương pháp xây dựng được phát triển để tìm ra kết nối gần nhất giữa bất kỳ hai cụm liên kề nào, đảm bảo tuyến đường liên kết các cụm cũng là ngắn nhất. Thuật toán K-means còn kết hợp với thuật toán tìm kiếm theo chiều sâu (DFS) để lập bản đồ thông qua phân chia khu vực quan tâm (ROI) thành các không gian nhỏ hơn [137]. Trong khi thuật toán K-means phân vùng các điểm lấy mẫu ở gần nhau dựa trên khoảng cách Euclide của chúng, thuật toán DFS xác định thứ tự di chuyển giữa các cụm. Thuật toán K-means kết hợp phân cụm Canopy, là một bước tiền xử lý để cải thiện hiệu quả của thuật toán K-means trong việc phân vùng không gian chưa biết. Cụ thể, phân cụm Canopy sử dụng hai ngưỡng khoảng cách và một thước đo khoảng cách xấp xỉ để nhanh chóng chia dữ liệu thành các tập con chồng lấn nhau [138]. Các điểm nằm quá gần tâm, trong bán kính gần sẽ bị loại bỏ khỏi các lần xử lý tiếp theo, trong khi các điểm nằm giữa cụm canopy được thêm vào cụm. Thuật toán K-means chỉ cần tính toán khoảng cách giữa các điểm nằm trong cùng một tập con, qua đó giảm đáng kể thời gian tính toán.

Mặc dù các phương pháp phân cụm cứng đã khảo sát có những ưu điểm riêng, chúng đều tồn tại một hạn chế cốt lõi: tính cứng nhắc trong ranh giới phân chia. Việc quy định ranh giới rõ ràng, không chồng lấn gây ra khó khăn lớn khi ứng dụng trong môi trường động và phức tạp của hệ thống đa robot. Thực tế này đặt ra yêu cầu cấp thiết về một cách tiếp cận "mềm" trong phân cụm, cho phép mô hình hóa sự phân bố dữ liệu theo xác suất và chấp nhận sự chồng lấn giữa các vùng quản lý. Trong bối cảnh đó, mô hình GMM và kỹ thuật phân cụm mềm đã vượt qua vai trò của công cụ phân tích dữ liệu thống kê thuần túy để trở thành một nền tảng toán học cốt lõi trong nhiều ứng dụng robot khác. Cụ thể, các nghiên cứu gần đây đã khai thác GMM trong việc học và mã hóa các kỹ năng vận động phức tạp, từ các thao tác của cánh tay robot [139] [140] [141] [142] đến các hành động cụ thể như gấp/rót [143] [144] [145]. Bên cạnh đó, GMM cũng được sử dụng hiệu quả để điều hướng trong môi trường phức tạp [146] [147] hoặc tìm kiếm, khám phá trong không gian rộng [148], áp dụng cho robot di động hoặc UAV. Các nghiên cứu này thường tập trung vào việc tối ưu hóa tham số mô hình để cân bằng giữa độ chính xác và khả năng tổng quát hóa, thường kết hợp với các kỹ thuật lai để khắc phục nhược điểm về khởi tạo; tái tạo quỹ đạo, mô hình hoá môi trường và tương tác thông qua phân cụm.

1.5 Thách thức và khoảng trống nghiên cứu

Các khảo sát đã cho thấy bức tranh đa dạng về các phương pháp hoạch định đường đi, từ cổ điển đến mô phỏng sinh học, việc áp dụng chúng vào các kịch bản MRPP trong thế giới thực vẫn còn đối mặt với nhiều thách thức đáng kể. Một trong những thách thức cơ bản nhất nằm ở sự cân bằng giữa chất lượng tối ưu và khả năng mở rộng. Phương pháp cổ điển như A* tuy đảm bảo tính tối ưu về đường đi nhưng lại trở nên kém hiệu quả khi số chiều không gian tăng lên, cần chi phí tính toán rất lớn với không gian cấu hình nhiều chiều. Phương pháp phỏng đoán đem lại sự linh hoạt nhưng thường xuyên đối mặt với nguy cơ mắc kẹt tại các điểm tối ưu cục bộ; phương pháp APF cũng gặp phải vấn đề cố hữu này. Thách thức càng trở nên nghiêm

trọng khi quy mô hệ thống tăng, bởi độ phức tạp của bài toán MRPP gia tăng đáng kể theo số lượng phần tử robot trong hệ thống.

Thách thức lớn thứ hai là việc xử lý môi trường động và tính không chắc chắn. Hầu hết các phương pháp hoạch định toàn cục đều giả định môi trường đã biết trước và không đổi, khiến chúng không thể đối phó với những tình huống không dự đoán trước. Mặc dù các kỹ thuật tránh va chạm cục bộ được đề xuất để giải quyết vấn đề này, nhưng bản thân chúng lại có những hạn chế riêng, ví dụ như APF bị dao động.

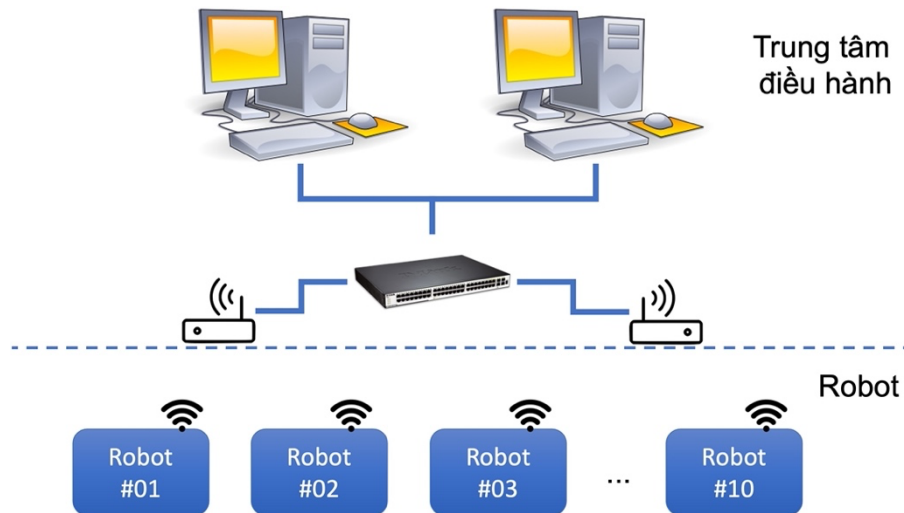
Từ những đánh giá, phân tích như trên, nghiên cứu sinh đề xuất thực hiện những nội dung nghiên cứu theo định hướng tập trung giải quyết thách thức thứ nhất, hình thành cùng với sự phát triển rất nhanh chóng của hệ thống đa robot có quy mô ngày càng lớn. Về đối tượng nghiên cứu, nghiên cứu sinh tập trung vào hệ thống đa robot với kiến trúc điều khiển tập trung, hoạch định đường đi và phân cụm các phần tử robot. Trong đó, hệ thống đa robot bao gồm tập hợp nhiều phần tử có cấu tạo tương tự nhau với hình dạng phổ biến là khối hộp chữ nhật, số lượng phần tử ≥ 10 ; hoạt động được quản lý, giám sát từ trạm điều khiển trung tâm.

- Mỗi phần tử robot được phân công tác một tác vụ, tương ứng với vị trí bắt đầu và vị trí đích đến cụ thể. Các tác vụ độc lập với nhau và có vị trí bắt đầu và vị trí đích đến khác nhau.

- Hoạch định đường đi được thực thi ở trạm điều khiển trung tâm để xác định được hành trình di chuyển cho từng phần tử và truyền dữ liệu hành trình xuống từng phần tử robot.

- Phần tử robot thực thi các chức năng điều khiển cục bộ như: ổn định tốc độ, bám theo quỹ đạo, tự động phát hiện và tránh vật cản... Một số cơ chế tránh va chạm điển hình có thể được áp dụng là: DWA (cửa sổ động) tích hợp trong ROS/ROS2 dựa vào bộ thông số vận tốc, gia tốc, dữ liệu về vật cản và quỹ đạo đã thiết lập; TEB (co dẫn thời gian) tích hợp trong ROS/ROS2 với nguyên lý gần tương tự như APF. Ngoài ra, một số thuật toán ứng dụng khác như APF, VO...

- Trong quá trình hoạt động, các robot chia sẻ thông tin về trạng thái hoạt động, dữ liệu vận hành (vị trí, hướng di chuyển, tốc độ...) với trạm điều khiển trung tâm. Ngược lại, dữ liệu vận hành của hệ thống được trạm điều khiển trung tâm gửi xuống các robot để cung cấp thông tin hỗ trợ cho điều khiển cục bộ tại từng phần tử.



Hình 1.6: Kiến trúc điều khiển của một hệ thống đa robot

Mô hình điều khiển được phân chia thành hai cấp, tương ứng với kiểu kiến trúc của một hệ thống đa robot minh họa như Hình 1.6, cấp điều khiển hệ thống và cấp điều khiển tại chỗ. Trong đó, cấp điều khiển hệ thống thực thi những tính năng chung cho toàn bộ hệ thống như phân công tác vụ khác nhau cho từng robot, tương ứng với điểm bắt đầu và điểm đích đến cần đạt; hoạch định đường đi tối ưu giữa điểm bắt đầu và điểm đích đến để thiết lập hành trình cho từng robot. Các phần tử robot có khả năng liên lạc và chia sẻ thông tin qua kênh truyền thông. Cấp điều khiển tại chỗ liên quan trực tiếp đến thực thi điều khiển chuyển động của từng phần tử robot căn cứ theo hành trình đã được thiết lập từ trước; tránh vật cản và những hành động trực tiếp khác nằm trong phạm vi quyền hạn của từng phần tử.

Về các hướng nghiên cứu, thứ nhất là nghiên cứu giải pháp hoạch định đường đi cho hệ thống đa robot dựa trên nền tảng trí tuệ tập thể, trọng tâm là thuật toán Tối ưu bầy ong. Giải quyết bài toán giới hạn của không gian tìm kiếm cố định trong thuật toán ABC truyền thống. Nghiên cứu sinh tập trung nghiên cứu theo định hướng phát triển cơ chế biểu diễn giải pháp linh hoạt, cho phép thuật toán tự động điều chỉnh cấu

trúc và độ phức tạp của đường đi tương thích với môi trường. Mục tiêu là xây dựng một mô hình kết hợp hiệu quả: vừa tận dụng khả năng tìm kiếm toàn cục của bầy đàn để tối ưu hóa chi phí di chuyển, vừa tích hợp khả năng biến đổi cấu trúc cục bộ để thích ứng nhanh với các vật cản trong quá trình vận hành. Định hướng này góp phần đáp ứng mục tiêu nghiên cứu thứ nhất của Luận án.

Thứ hai là giải quyết vấn đề áp lực tính toán và khả năng phối hợp khi quy mô hệ thống đa robot tăng cao, nghiên cứu đề xuất giải pháp phân chia không gian làm việc thông qua ứng dụng năng lực "phân cụm mềm" của GMM để phân chia các phần tử robot vào các cụm hoạt động và chia sẻ dữ liệu vận hành hệ thống một cách độc lập, giải quyết bài toán tổ chức và phối hợp cho hệ thống đa robot. Thiết lập cơ chế cân bằng về số lượng phần tử robot trong từng cụm thông qua hàm tiêu toàn phương. Định hướng này góp phần đáp ứng mục tiêu nghiên cứu thứ hai của Luận án.

1.6 Kết luận chương 1

Chương 1 đã tổng kết đầy đủ về sự phát triển cùng thực trạng nghiên cứu của MRS. Sự chuyển dịch từ một phần tử robot đơn lẻ sang các hệ thống phức tạp đã giúp MRS trở thành giải pháp then chốt trong các lĩnh vực hậu cần, nông nghiệp và cứu nạn, cho phép các phần tử robot phối hợp chia sẻ nhiệm vụ mà một cá thể đơn lẻ không thể hoàn thành. Với bối cảnh như vậy, MRPP cũng có xu hướng dịch chuyển rõ rệt từ thuật toán cổ điển sang phương pháp trí tuệ tập thể và mô phỏng sinh học. Trong đó, thuật toán ABC được đánh giá cao nhờ tính đơn giản và khả năng tìm kiếm toàn cục mạnh mẽ, tuy nhiên vẫn tồn tại hạn chế về tốc độ hội tụ và khả năng khai thác cục bộ ở giai đoạn sau của quá trình tìm kiếm. Đồng thời, để quản lý hiệu quả hệ thống khi quy mô số lượng tăng lên, các kỹ thuật phân cụm đã được khảo sát, đặc biệt nhấn mạnh sự cần thiết của các phương pháp phân cụm "mềm" như mô hình hỗn hợp Gauss (GMM) để thay thế tính cứng nhắc của các ranh giới phân chia truyền thống và xử lý tốt hơn tính không chắc chắn trong môi trường biến động. Mặc dù các công trình hiện có đã đạt được nhiều thành tựu, nhưng thách thức về sự cân bằng giữa chất lượng tối ưu và khả năng mở rộng vẫn là một khoảng trống nghiên cứu lớn cần

giải quyết. Từ đó, luận án xác định hướng nghiên cứu tập trung vào nghiên cứu thuật toán ABC với cải tiến về cấu trúc và cơ chế biểu diễn linh hoạt và ứng dụng năng lực phân cụm của GMM để thiết lập cơ chế điều phối hiệu quả cho hệ thống đa robot quy mô lớn , góp phần đáp ứng trực tiếp hai mục tiêu nghiên cứu trọng tâm của đề tài.

Chương 2. TỐI ƯU BẦY ONG TRONG HOẠCH ĐỊNH ĐƯỜNG ĐI

Qua việc tìm hiểu và khảo sát những nghiên cứu về thuật toán ABC, những cải tiến cho ABC đã tập trung vào hai hướng chính. Thứ nhất là kết hợp với những thuật toán khác, như EP, ACO, hoặc phương pháp học (TLBO). Thứ hai là đề xuất những thay đổi trong cấu trúc vận hành của thuật toán, như cải thiện sự cân bằng giữa thăm dò và khai thác. Luận án đề xuất một giải pháp cải tiến mới cho thuật toán ABC, biến thể ElasticABC, tập trung giải quyết vấn đề khả năng khai thác cục bộ hạn chế và không gian tìm kiếm cố định. Khi đó, số lượng điểm trung gian trong hành trình của từng phần tử robot được thích nghi theo điều kiện môi trường. Nội dung của chương này trình bày chi tiết về quá trình xây dựng thuật toán với cặp toán tử đối ngẫu Co và Dẫn, khởi tạo phỏng đoán, kỹ thuật xử lý chuyên dụng DeZigzagPath, các phương trình cho ba giai đoạn thực thi của thuật toán. Bên cạnh đó, một số khung thuật toán cơ bản của ElasticABC cũng được giới thiệu trong chương này. Nội dung trình bày của Chương 2 góp phần đáp ứng mục đích nghiên cứu thứ nhất của Luận án.

2.1 Đặt vấn đề

Hoạch định đường đi là một trong những bài toán cốt lõi và thách thức nhất trong lĩnh vực robot tự hành. Mục tiêu không chỉ là tìm kiếm đường đi từ điểm đầu đến điểm cuối không va chạm, mà còn tối ưu hóa các yếu tố như độ dài đường đi và thời gian tìm kiếm, đặc biệt trong các môi trường phức tạp. Phương pháp truyền thống có hạn chế về khả năng mở rộng và chi phí tính toán khi không gian cấu hình trở nên lớn và phức tạp. Để giải quyết vấn đề này, các phương pháp tối ưu dựa trên trí tuệ tập thể đã được áp dụng rộng rãi nhờ khả năng tìm kiếm ngẫu nhiên thông minh và tính linh hoạt cao. Trong số đó, ABC được đánh giá cao nhờ cấu trúc đơn giản, dễ cài đặt và khả năng tìm kiếm toàn cục mạnh mẽ.

Tuy nhiên, như mọi thuật toán siêu phỏng đoán khác, ABC tồn tại nhược điểm cố hữu như khả năng khai thác cục bộ chưa thực sự hiệu quả, làm cho tốc độ hội tụ

chậm trong giai đoạn sau của quá trình tìm kiếm. Việc áp dụng trực tiếp ABC vào bài toán hoạch định đường đi đòi hỏi những điều chỉnh và cải tiến phù hợp để đảm bảo sự cân bằng giữa khả năng tìm kiếm đa dạng và khả năng hội tụ nhanh về giải pháp tối ưu. Cụ thể, nếu thuật toán ABC truyền thống giới hạn không gian tìm kiếm trong các véc-tơ có số chiều cố định thì biến thể ElasticABC, lựa chọn cách tiếp cận hoàn toàn khác biệt. Theo đó, đường đi của phần tử robot được mô tả bằng cơ chế biểu diễn linh hoạt: không là tập hợp cố định các điểm trung gian (waypoint) mà được mô hình hóa như là chuỗi co dãn, có khả năng tự động thêm điểm để vòng tránh vật cản và bớt điểm để tối ưu hóa số lượng điểm trung gian. Đây là cách tiếp cận tự thích nghi cấu trúc của giải pháp theo độ phức tạp của môi trường và không phụ thuộc vào tham số kích thước quần thể cố định. ElasticABC có ba đóng góp cụ thể như sau:

i. Cải tiến thuật toán ABC truyền thống với cốt lõi là tối ưu cấu trúc, sử dụng cặp toán tử đối ngẫu Co và Dẫn hoạt động song song trong giai đoạn tìm kiếm của ong quan sát, khắc phục nhược điểm về khả năng khai thác cục bộ thông qua cơ chế co dãn cho chuỗi điểm trung gian của đường đi robot. Trong đó,

- Toán tử Dẫn đóng vai trò đảm bảo tính khả thi của đường đi, khi một đoạn đường đi bị phát hiện vi phạm vùng an toàn của vật cản, tự động thêm điểm trung gian để mở rộng không gian tìm kiếm, giúp điều chỉnh hướng đường đi tránh khỏi khu vực kẹt

- Toán tử Co đóng vai trò bộ lọc, triệt tiêu các điểm trung gian dư thừa ngay khi đường đi đảm bảo an toàn, để ép đường đi hội tụ nhanh về quỹ đạo ngắn nhất.

ii. Tích hợp chiến lược khởi tạo phỏng đoán dựa trên toán tử Dẫn, thay thế cho hình thức khởi tạo ngẫu nhiên trong thuật toán ABC truyền thống. Chất lượng đầu vào của giai đoạn tìm kiếm được cải thiện tốt hơn khi lựa chọn giải pháp ban đầu có chủ đích, qua đó cải thiện tốc độ hội tụ đến giải pháp gần tối ưu ở giai đoạn tiếp theo.

iii. Tích hợp kỹ thuật xử lý chuyên dụng (DeZigzagPath) để xử lý hiện tượng gấp khúc cục bộ, hoàn thiện chất lượng đường đi thực tế.

Tính mới của ElasticABC thể hiện trong sự chuyển dịch từ tối ưu hóa tham số sang tối ưu hóa cấu trúc, số lượng điểm trung gian được xác định theo cơ chế biểu diễn linh hoạt căn cứ theo đặc điểm thực tế của không gian làm việc và quỹ đạo di chuyển cần thiết lập cho từng phần tử robot. Nói cách khác, đường đi của robot thích nghi một cách có hiệu quả với độ phức tạp của môi trường thông qua việc điều chỉnh có chọn lọc số lượng điểm trung gian. ElasticABC đạt được sự cân bằng giữa khả năng khám phá và khả năng khai thác. Ngoài ra, ba điểm i, ii và iii đều được tích hợp hoàn chỉnh trong một thuật toán duy nhất.

2.2 Mô hình hoá không gian làm việc

Xét một không gian làm việc cho N robot được bố trí các vật cản tĩnh. Không gian có kích thước xác định và được biểu diễn bởi biểu thức sau:

$$\mathcal{W} = \mathbb{R}^n \quad (2.1)$$

trong đó $n = 2$ hoặc 3 , số chiều của không gian làm việc.

Định nghĩa không gian vật cản $\mathcal{O}$ thuộc $\mathcal{W}$, có chứa các vật cản tĩnh. Một vật cản tĩnh có vị trí cố định không thay đổi theo thời gian kể từ thời điểm nó xuất hiện trong không gian làm việc. Giả thiết $\mathcal{W}$ có m vật cản, $\mathcal{O}$ có thể được xác định bởi công thức sau:

$$\mathcal{O} = \mathcal{O}_1 \cup \mathcal{O}_2 \dots \cup \mathcal{O}_m \subset \mathcal{W} \quad (2.2)$$

trong đó $\mathcal{O}_1, \mathcal{O}_2, \dots, \mathcal{O}_m$ lần lượt là không gian của từng vật cản. Không gian tự do $\mathcal{F}$ là một phần cụ thể của không gian làm việc $\mathcal{W}$ nơi các robot có thể di chuyển tự do. Chúng ta có mối quan hệ giữa $\mathcal{F}$ và $\mathcal{W}$ như sau:

$$\mathcal{F} \subset \mathcal{W} \quad (2.3)$$

và

$$(\mathcal{F} \cup \mathcal{O}) \subset \mathcal{W} \quad (2.4)$$

Biểu thức (2.4) cho thấy mối liên hệ giữa kích thước robot đến vùng không gian có thể tiếp cận tại một số vùng thuộc $\mathcal{W}$ và không thuộc $\mathcal{O}$.

Khi N phần tử robot, được gọi là $A_1, A_2, \dots, A_N$, di chuyển theo các quỹ đạo độc lập từ điểm xuất phát riêng đến các đích trong không gian làm việc $\mathcal{W}$. Giả thiết tập hợp điểm xuất phát và tập hợp điểm đích đến của N robot lần lượt là

$$(q_{10}, q_{20}, \dots, q_{N0}) \quad (2.5)$$

và

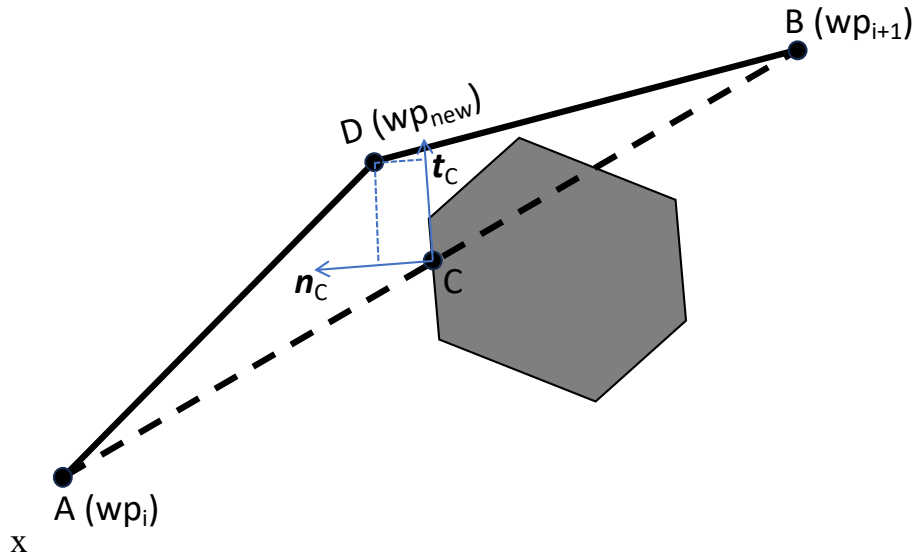
$$(q_{1G}, q_{2G}, \dots, q_{NG}) \quad (2.6)$$

Các điểm đều trong không gian tự do $\mathcal{F}$, quỹ đạo của N robot được biểu diễn như sau:

$$\begin{bmatrix} Path_1(t) \\ \vdots \\ Path_N(t) \end{bmatrix}$$

trong đó quỹ đạo của A_r được định nghĩa như sau:

$$Path_r(t) = f_{pp}(q_{ro} \quad q_{rG}) \quad (2.7)$$



Hình 2.1: Nguyên lý thêm điểm của toán tử Dẫn

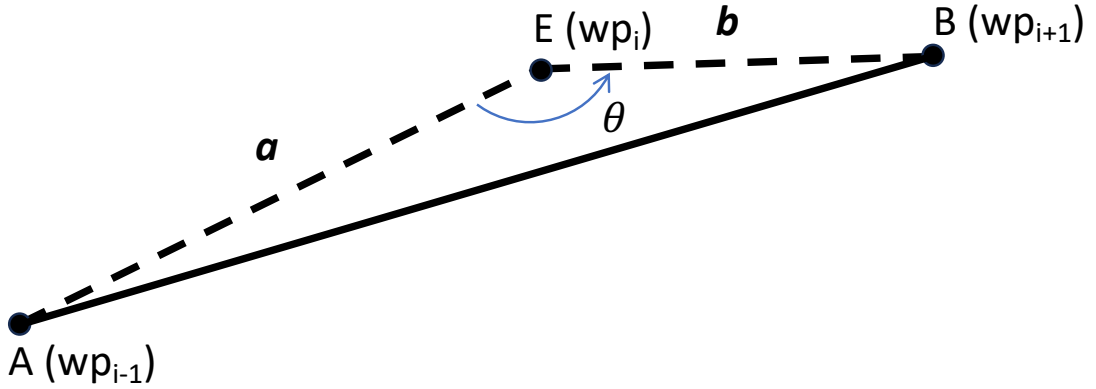
2.3 Cơ chế co dẫn với cặp toán tử đối ngẫu

Toán tử Dẫn được dùng khi một đoạn hành trình của phần tử robot cắt qua vật cản; Hình 2.1 minh họa phương án thêm điểm của toán tử Dẫn, được xét trong tình huống tổng quát vật cản có biên dạng lồi. Điểm trung gian được thêm mới được xác

định theo thành phần pháp tuyến và thành phần tiếp tuyến của bề mặt tại điểm va chạm; toạ độ của nó được xác định theo biểu thức như sau

$$wp_{new} = C + (d_{safe} + \epsilon) \cdot \mathbf{n}_C + \lambda \cdot \mathbf{t}_C \quad (2.8)$$

trong đó C là toạ độ của điểm va chạm; d_{safe} là khoảng an toàn tối thiểu để robot không va chạm với vật cản khi di chuyển; $\epsilon \geq 0$ là hệ số ngẫu nhiên để duy trì đặc điểm khám phá toàn cục trong giai đoạn khởi tạo, $\mathbf{n}_C$ là vec-tơ pháp tuyến đơn vị; λ là hệ số trượt theo phương tiếp tuyến, nhận giá trị ngẫu nhiên, $\mathbf{t}_C$ là vec-tơ tiếp tuyến đơn vị.



Hình 2.2: Nguyên lý bớt điểm của toán tử Co

Toán tử Co được sử dụng để loại bỏ bớt điểm trung gian trong bộ ba điểm liên tiếp gần thẳng hàng với nhau. Hình 2.2 minh hoạ phương án bớt điểm của toán tử Co thông qua kiểm tra góc mở θ tại điểm trung gian nằm giữa, biểu thức xác định là

$$\cos \theta = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \|\mathbf{b}\|} \quad (2.9)$$

với $\mathbf{a}$ là vectơ đoạn nối từ $wp_{r,i-1}$ đến $wp_{r,i}$; $\mathbf{b}$ là vectơ đoạn nối từ $wp_{r,i}$ đến $wp_{r,i+1}$. Góc mở θ này phải vượt quá ngưỡng cho phép để làm điều kiện xoá điểm $wp_{r,i}$; đồng thời việc xoá không tạo ra đường giao cắt vật cản mới.

Thuật toán 1: Lớp Co dẫn với các toán tử thành phần

`classdef ElasticOperators`

`function InsertOp`

`% Hàm gọi toán tử Dẫn`

	function <i>DeleteOp</i>	% Hàm gọi toán tử Dẫn
	function <i>DeZigzagPath</i>	% Hàm gọi toán tử khử gấp khúc
	<i>Toán tử Dẫn</i>	
	Đầu vào: Đường đi cần kéo dẫn $path(i)$; thông tin điểm va chạm, tọa độ C , vec-tơ pháp tuyến nC	
	Đầu ra: Đường đi cập nhật $path(i)$	
1	BẮT ĐẦU	
2	Khởi tạo thông số của toán tử Dẫn: $\epsilon, \lambda, d_{safe}$	
3	Tìm tọa độ điểm trung gian mới theo công thức (2.8)	
4	Cập nhật đường đi $path(i)$	
5	KẾT THÚC	
	<i>Toán tử Co</i>	
	Đầu vào: Đường đi cần co lại $path(i)$; mô hình hệ thống $model$	
	Đầu ra: Đường đi cập nhật $path(i)$	
1	BẮT ĐẦU	
2	Khởi tạo thông số của toán tử Co: $cos\Theta$	
3	Xác định tọa độ của bộ ba điểm trung gian liên tiếp từ $path(i)$	
4	Tính góc mở giữa hai đoạn nối từng cặp điểm trung gian theo công thức (2.9)	
5	NẾU góc mở $> cos\Theta$ THÌ xóa điểm trung gian nằm giữa	
6	NẾU đoạn nối cặp điểm trung gian mới không gây va chạm THÌ cập nhật đường đi $path(i)$	
7	KẾT THÚC	

2.4 Biểu diễn đường đi và khởi tạo phỏng đoán

Xét một hệ thống đa robot với N phần tử, tập hợp điểm xuất phát và tập hợp đích đến các phần tử robot được quy ước lần lượt là (2.5) và (2.6). Khi đó, biểu thức (2.7) mô tả quỹ đạo của phần tử A_r được viết lại như sau

$$Path_r(t) = f_{pp}(q_{r0}, wp_{r,1} \dots wp_{r,i} \dots wp_{r,k}, q_{rG}) \quad (2.10)$$

trong đó $wp_{r,i}$ là toạ độ điểm trung gian thứ i , k là số lượng điểm trung gian.

Đường đi của robot A_r chỉ bao gồm điểm bắt đầu q_{r0} và điểm đích đến q_{rG} ; các điểm trung gian được thêm vào theo hoạt động của toán tử Dẫn. Trong biểu thức (2.8), thành phần ngẫu nhiên ϵ làm cho giải pháp khởi tạo không có tính chất lặp lại, đảm bảo được yếu tố khám phá cho khởi tạo quần thể ban đầu.

Thuật toán 2: Khởi tạo phỏng đoán cho ElasticABC

Đầu vào: mô hình hệ thống *model*; mô hình quần thể *Pop*

Đầu ra: Đường đi $path(i)$

- 1 **BẮT ĐẦU**
 - 2 Khởi tạo đường đi $path(i)$
 - 3 VÒNG LẶP:
 - 4 Gọi toán tử Dẫn (*ElasticOperators.InsertOp*)
 - 5 Cập nhật điểm trung gian trong đường đi $path(i)$
 - 6 NẾU vòng lặp $\geq$ giới hạn cho phép THÌ chuyển sang 8
 - 7 NẾU đoạn nối cặp điểm trung gian gây va chạm THÌ quay lại 3
 - 8 Gọi toán tử Co (*ElasticOperators.DeleteOp*) để loại bỏ bớt điểm trung gian
 - 9 Tính toán giá trị chi phí của đường đi $path(i)$
 - 10 NẾU có đoạn nối điểm trung gian gây va chạm THÌ giá trị chi phí nhận phạt 100
 - 11 **KẾT THÚC**
-

2.5 Thiết lập hàm mục tiêu

Mặc dù sử dụng cơ chế Co Dẫn đã kiểm soát được vấn đề an toàn va chạm với vật cản tĩnh, hàm mục tiêu được thiết lập bao gồm tổng độ dài Euclid và thành phần phạt va chạm. Xét cho phân tử robot A_r , hàm mục tiêu được mô tả như sau

$$f_{obj,r} = L_r + \omega_1 \sum_{j=1}^m S(r,j) \quad (2.11)$$

trong đó L_r là khoảng cách Euclid giữa hai điểm trung gian thứ $i+1$ và i , $wp_{r,i+1}$ và $wp_{r,i}$

$$L_r = \sum_{i=0}^k \|wp_{r,i+1} - wp_{r,i}\|_2 \quad (2.12)$$

ω_1 là hệ số phạt, có giá trị đủ lớn để ngăn phần tử va chạm với vật cản; $S(r,j)$ là hàm kiểm tra va chạm được xác định theo biểu thức sau

$$S(i,j) = \begin{cases} 1, & \text{if } wp_{r,j} \in \mathcal{O} \\ 0, & \text{if } wp_{r,j} \notin \mathcal{O} \end{cases} \quad (2.13)$$

trong đó $wp_{r,j}$ là điểm trung gian thứ j của phần tử A_r .

2.6 Ba giai đoạn của quần thể ong

Thuật toán ABC được thực thi theo vòng lặp với ba giai đoạn, biểu thức toán học được mô tả căn cứ theo công trình của Karaboga và các cộng sự [61] [62]. Trong giai đoạn ong thợ (giai đoạn 1), mỗi cá thể sẽ tìm kiếm giải pháp mới, ở đây chính là vị trí mới cho điểm trung gian trong vùng lân cận của vị trí hiện tại $wp_{i,j}$; khi đó vị trí mới được xác định theo phương trình cập nhật vị trí điểm trung gian thứ j của cá thể i là

$$wp_{i,j}^{new} = wp_{i,j} + \phi_{i,j}(wp_{i,j} - wp_{l,j}) + \psi_{i,j}(wp_{best,j} - wp_{i,j}) \quad (2.14)$$

trong đó $wp_{l,j}$ là tọa độ điểm trung gian thứ j của cá thể lân cận l ngẫu nhiên và $l \neq i$, $wp_{best,j}$ là tọa độ điểm trung gian thứ j của cá thể tốt nhất toàn cục, được thêm vào để tăng tính định hướng; $\phi_{i,j}$ là hệ số ngẫu nhiên trong đoạn $[-1, 1]$, $\psi_{i,j}$ là hệ số định hướng, được chọn trong khoảng $(0, 1)$. Toán tử Dẫn và toán tử Co được dùng để chỉnh sửa cấu trúc nếu điểm trung gian mới tạo ra đoạn đường có va chạm hoặc vi phạm điều kiện dư thừa điểm trung gian. Cuối cùng, tọa độ điểm trung gian mới được giữ lại nếu giá trị thích nghi của nó tốt hơn so với vị trí hiện tại.

Trong giai đoạn ong quan sát (giai đoạn 2), xác suất chọn giải pháp được xác định theo biểu thức sau

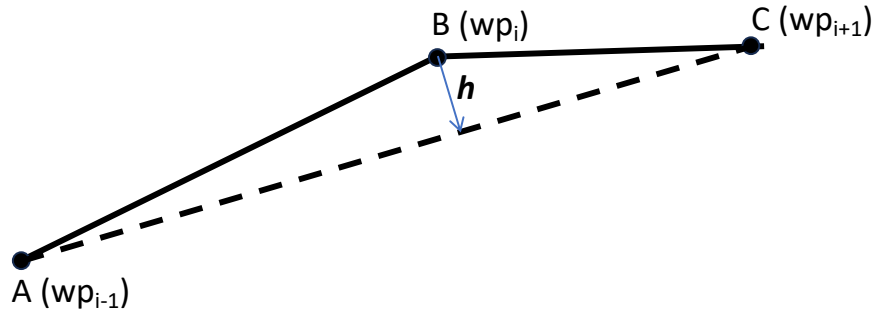
$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \quad (2.15)$$

trong đó fit_i là hàm thích nghi, được xác định theo biểu thức sau

$$fit_i = \frac{1}{1 + f_{obj,i}} \quad (2.16)$$

còn $f_{obj,i}$ là hàm mục tiêu và được xác định theo công thức (2.11).

Trong giai đoạn ong trình sát (giai đoạn 3), nếu một giải pháp không được cải thiện sau một số chu kỳ được xác định trước, nó sẽ bị loại bỏ. Con ong thợ liên quan trở thành ong trình sát và khám phá một giải pháp mới bằng cách sử dụng khởi tạo phỏng đoán thông qua công thức (2.10) kết hợp với toán tử Dẫn, công thức (2.8) để tạo ra một giải pháp mới.



Hình 2.3: Kỹ thuật khử gấp khúc

2.7 Kỹ thuật DeZigzagPath

Kỹ thuật khử đoạn gấp khúc, DeZigzagPath, được thiết kế như một công cụ đuổi thẳng đoạn gấp khúc cục bộ trong đường đi của robot trong khi vẫn phải bảo toàn được đặc trưng hình học chủ đạo đã tạo ra bởi thuật toán ElasticABC. DeZigzagPath sử dụng tham số biên độ "dao động" cho phép làm tiêu chuẩn phân loại đoạn gấp khúc. Cụ thể hơn, một thuật toán lặp cho bộ ba điểm trung gian liên tiếp, tính toán khoảng cách từ điểm hiện tại đến đoạn thẳng nối tắt giữa hai điểm trước và sau h , minh họa như Hình 2.3. Điểm trung gian có giá trị $h \leq maxAmp$ sẽ bị loại bỏ với

$maxAmp$ là giá trị ngưỡng biên độ cho phép. Đoạn nối tắt mới cũng được kiểm tra an toàn, đảm bảo không cắt qua chướng ngại vật.

2.8 Quy trình thực hiện ElasticABC

- | | |
|-------------|--|
| Khởi tạo | Khởi tạo phỏng đoán quần thể kết hợp dùng toán tử Dẫn, theo biểu thức (2.8), để thêm điểm trung gian cho phương trình biểu diễn (2.10). Thông qua hệ số ngẫu nhiên, quần thể tạo ra đảm bảo được sự đa dạng hoá. |
| Giai đoạn 1 | Giai đoạn ong thợ, thực hiện tìm kiếm vị trí điểm trung gian mới theo phương trình (2.14), các toán tử Dẫn (2.8) và toán tử Co (2.9) được dùng để chỉnh sửa cấu trúc (nếu cần thiết). |
| Giai đoạn 2 | Giai đoạn ong quan sát, chọn giải pháp theo giá trị xác suất (2.15) của hàm thích nghi (2.16). |
| Giai đoạn 3 | Giai đoạn ong trinh sát, loại bỏ giải pháp nếu không cải thiện sau số lượng vòng lặp nhất định. |

Thuật toán 3: Thuật toán hoạch định đường đi ElasticABC

- 1 **BẮT ĐẦU**
- 2 Khởi tạo môi trường: số lượng robot, kích thước bản đồ, vị trí vật cản
 $model.map(N_ROBOTS, MAP_SIZE, OBS_DENSITY, OBS_DIM)$
- 3 Khởi tạo quần thể, số vòng lặp và giá trị giới hạn: $Pop, MaxIt, Limit$
- 4 Gọi Thuật toán 2: Khởi tạo phỏng đoán cho ElasticABC
- 5 VÒNG LẶP:
 % Giai đoạn ong thợ
- 6 Tìm kiếm lân cận theo công thức (2.14)
- 7 Kiểm tra va chạm với điểm trung gian mới tìm được

- 8 Cập nhật giá trị chi phí theo cơ chế tham lam, cập nhật biến giới hạn thử *Pop.trial*
% Giai đoạn ong quan sát
 - 9 Tính toán xác suất lựa chọn theo giá trị thích nghi, công thức (2.15) và (2.16)
 - 10 Chọn giải pháp tốt theo cơ chế vòng quay *Roulette*
 - 11 Kiểm tra va chạm với giải pháp được chọn
 - 12 Cập nhật giải pháp tốt nhất và giá trị chi phí
% Giai đoạn ong trình sát
 - 13 NẾU *Pop.trial* > *Limit* THÌ xoá bỏ *path* và gọi Thuật toán 2: Khởi tạo phỏng đoán cho ElasticABC
NẾU vòng lặp < *MaxIt* THÌ quay lại 5
 - 14 Xuất đường đi tốt nhất *path* và giá trị chi phí cuối cùng
 - 15 KẾT THÚC
-

2.9 Kết luận chương 2

Chương này đã trình bày chi tiết về quá trình xây dựng biến thể ElasticABC, cải tiến từ thuật toán ABC truyền thống, để hoạch định đường đi cho hệ thống đa robot với hai đặc điểm quan trọng. Thứ nhất về phương pháp luận, ElasticABC cải tiến nhược điểm của cấu trúc véc-tơ cố định trong phương pháp ABC truyền thống. Thông qua việc áp dụng cơ chế co giãn cùng cặp toán tử đối ngẫu kết hợp với khởi tạo phỏng đoán, thuật toán không những giảm thiểu không gian tìm kiếm không cần thiết ban đầu mà còn cho phép đường đi tự thích nghi với độ phức tạp của môi trường. Đồng thời, hoạt động song song của cặp toán tử Co và Dẫn đã tạo ra sự cân bằng động giữa khả năng khám phá, tránh vật cản hoặc thoát khỏi bẫy cục bộ, với khả năng khai thác, tối ưu hóa độ dài, loại bỏ điểm dư thừa. Thứ hai là đề xuất kỹ thuật xử lý hậu kỳ DeZigzagPath, giữ vai trò quan trọng trong việc hoàn thiện chất lượng đường đi; loại bỏ hiện tượng gấp khúc cục bộ trong khi vẫn giữ nguyên cấu trúc tổng thể của đường đi.

Những công bố liên quan đến Chương 2 có: Công trình 2 trong Danh mục các công trình khoa học. Ngoài ra, nghiên cứu sinh đã nộp bản thảo Công trình 7 cho tạp chí Array thuộc nhà xuất bản Elsevier.

Chương 3. PHƯƠNG PHÁP PHÂN CỤM ĐỘNG

Khi quy mô ngày càng gia tăng về số lượng phần tử, vận hành hệ thống đa robot trở nên phức tạp hơn trong khi vẫn cần đảm bảo khả năng đáp ứng nhanh chóng. Bên cạnh vấn đề điều khiển cục bộ của từng cá thể, việc quản lý vận hành của hệ thống cần duy trì tốc độ xử lý và chia sẻ thông tin vận hành từ trạm điều khiển trung tâm xuống các phần tử robot trong hệ thống với kiến trúc điều khiển tập trung. Do đó, tìm kiếm một cơ chế phân cụm trở thành yêu cầu cần thiết để hỗ trợ cho hoạt động quản lý vận hành. Luận án đề xuất một phương pháp phân cụm mới dựa theo cách tiếp cận phân chia mềm của GMM, lựa chọn được cấu hình phân cụm có số lượng phần tử của từng cụm là cân bằng: khả năng cân bằng tải. Hệ thống có thể chia sẻ dữ liệu độc lập theo từng cụm, qua đó giảm khối lượng dữ liệu vận hành của hệ thống cần trao đổi. Nội dung của chương này trình bày chi tiết về quá trình xây dựng phương pháp phân cụm động (DCM) với sự kết hợp giữa mô hình GMM và thuật toán EM, đề xuất tham số chi phí δ_k và hàm mục tiêu bình phương tối thiểu f_{obj} . Bên cạnh đó, một số khung thuật toán cơ bản của DCM cũng được giới thiệu trong chương này. Nội dung trình bày của Chương 3 góp phần đáp ứng mục đích nghiên cứu thứ hai của Luận án.

3.1 Đặt vấn đề

Quản lý hệ thống đa robot với số lượng phần tử lớn là thách thức đáng kể, chủ yếu do sự phức tạp trong MRPP và phân công nhiệm vụ. Với các hệ thống quy mô lớn, mỗi robot thường được phân công tác vụ độc lập, có vị trí xuất phát và đích đến khác nhau. Giải pháp MRPP phải giải quyết đồng thời tìm kiếm đường đi tối ưu, về quãng đường hoặc thời gian, cho từng robot trong mối tương tác liên tục với các robot khác, và bắt buộc phải tránh va chạm. Đồng thời giữa MRPP và phân công tác vụ có mối liên hệ chặt chẽ với nhau, quyết định phân công tác vụ không hiệu quả có thể trực tiếp tạo ra bài toán MRPP gần như không thể giải quyết được, dẫn đến sụt giảm hiệu suất toàn hệ thống.

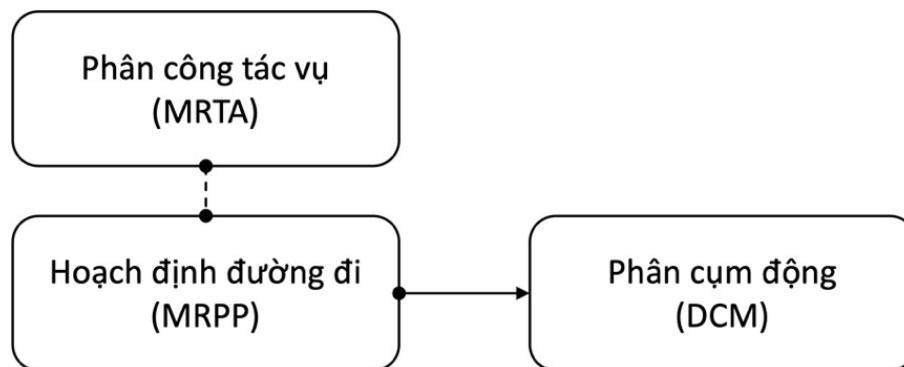
Một thách thức khác đến từ sự phức tạp của môi trường cùng với giới hạn về nhận thức. Môi trường vận hành quy mô lớn chứa nhiều vật cản, bao gồm vật cản tĩnh (bàn, ghế, giá hàng), vật cản động (con người, phương tiện khác), và những phần tử robot khác trong hệ thống (vật cản động đã được nhận thức). Điều này tạo ra áp lực lớn về nhận thức và tính toán. Hơn thế, với diện tích hoạt động lớn và biến động liên tục, cảm biến của từng phần tử riêng lẻ không đáp ứng đủ thông tin; đòi hỏi chúng phải dựa vào việc chia sẻ thông tin với hệ thống có nhận thức đầy đủ.

Hai giải pháp kỹ thuật được đề xuất tích hợp trong phần tử robot, hỗ trợ nhận thức môi trường xung quanh. Giải pháp thứ nhất là phương pháp ước lượng trạng thái động học của robot thông qua thiết bị đo lường quán tính và thuật toán xử lý dữ liệu thời gian thực. Dữ liệu đầu vào được tính toán để xác định các tham số trạng thái bao gồm: hướng định vị, vận tốc góc, gia tốc (tăng/giảm tốc), cũng như các biến động động lực học như sốc, rung và độ nghiêng. Các thông số ước lượng này đóng vai trò là đầu vào cho hệ thống điều khiển và giám sát, phát hiện các trạng thái bất thường nhằm đảm bảo tính an toàn và ổn định. Giải pháp thứ hai là một cơ cấu hiệu chỉnh hướng chuyên dụng cho hệ cảm biến phát hiện vật cản, cho phép điều chỉnh được FOV tổng thể như: mở rộng hoặc thu hẹp hoặc chệch về một phía (trái hoặc phải). Ưu điểm nổi bật của cơ cấu này là khả năng cấu hình một lần duy nhất, cho phép robot thích nghi với các hành lang có biên độ hẹp và cấu trúc chân giá hàng đa dạng mà không yêu cầu thiết kế lại hệ thống cơ khí.

Bên cạnh đó, hệ thống đa robot tồn tại một nghịch lý về hiệu suất liên quan đến quy mô các phần tử. Về mặt lý thuyết, việc gia tăng số lượng phần tử robot mang hai lợi ích rõ ràng. Thứ nhất là cải thiện nhận thức không gian thông qua khả năng thu thập dữ liệu nhiều hơn với tần suất nhanh hơn. Thứ hai là giảm tải công việc tổng thể của hệ thống với kỳ vọng hoàn thành nhiệm vụ nhanh hơn. Tuy nhiên, trên thực tế, sự gia tăng này lại tạo ra hai thách thức đối nghịch. Thứ nhất là chuyển động đồng thời của các phần tử lại tạo ra môi trường bất ổn và khó dự đoán hơn, đòi hỏi giám sát liên tục theo thời gian. Thứ hai là mật độ robot tăng cao làm gia tăng nguy cơ tắc nghẽn, khi các robot có thể cản trở lẫn nhau, thay vì đẩy nhanh tiến độ hoàn thành

nhiệm vụ. Do đó, thách thức then chốt không nằm ở việc tăng số lượng robot, mà ở khả năng duy trì nhận thức tình huống liên tục. Hệ thống phải đảm bảo rằng mỗi chuyển động riêng lẻ đều đóng góp tích cực vào hiệu quả tổng thể, thay vì vô tình tạo ra xung đột và làm tê liệt hệ thống.

Hệ quả của những thách thức này tập trung ở một vấn đề then chốt, đó là áp lực tính toán lên hệ thống. Khối lượng dữ liệu lớn, gia tăng theo quy mô và mức độ biến động môi trường buộc hệ thống phải giải quyết một mâu thuẫn cơ bản. Một mặt, việc tối ưu toàn cục mang lại hiệu suất tổng thể nhưng đòi hỏi tính toán tập trung, dễ quá tải và có độ trễ cao. Mặt khác, việc chỉ ra quyết định cục bộ tuy nhanh nhạy nhưng lại dễ gây ra tắc nghẽn ở quy mô lớn. Do đó, việc phân cụm robot thành các nhóm nhỏ, dễ quản lý hơn, là một giải pháp quan trọng. Đây chính là phương pháp trung gian cần thiết để cân bằng giữa yêu cầu tối ưu toàn cục và khả năng phản ứng linh hoạt cục bộ, qua đó giảm bớt áp lực tính toán cho hệ thống.



Hình 3.1: Liên hệ giữa DCM và MRPP

Cuối cùng, trên cơ sở phân tích và đánh giá những nghiên cứu hiện có, một phương pháp phân cụm động với hai giai đoạn sử dụng mô hình GMM được đề xuất cho hệ thống đa robot có kiến trúc điều khiển tập trung. Phương pháp này phân chia toàn bộ robot thành nhiều nhóm nhỏ, dễ quản lý hơn. Hình 3.1 minh họa mối quan hệ giữa DCM và lập kế hoạch đường đi cũng như phương pháp phân công tác vụ. DCM nhằm mục đích cách giảm khối lượng dữ liệu vận hành gửi từ trạm điều khiển trung tâm xuống bằng cách cân bằng tải cho từng cụm và dữ liệu chia sẻ cho từng cụm độc lập với nhau, qua đó giảm tải công việc tổng thể của hệ thống. Khi đó, mỗi robot chỉ

cần duy trì liên lạc và phối hợp hoạt động (nếu cần) với các robot khác trong nhóm được chỉ định.

3.2 Mô hình hoá dữ liệu đầu vào

Xét một hệ thống đa robot với N phần tử, tập hợp điểm xuất phát và tập hợp đích đến các phần tử robot được quy ước lần lượt là (2.5) và (2.6); biểu thức (2.7) mô tả quỹ đạo của phần tử A_r . Nếu tồn tại điểm giao cắt giữa quỹ đạo của A_r với quỹ đạo của A_i , chúng ta định nghĩa các điểm giao cắt như sau:

$$(Co_{r,i}) = Path_r(t) \cap Path_i(t) \neq \emptyset \quad (3.1)$$

Vì vậy, tập dữ liệu các điểm giao cắt giữa các quỹ đạo của N robot là:

$$C = \bigcup_{r,i=0}^N (Co_{r,i}) \quad (3.2)$$

và mối quan hệ giữa mỗi phần tử robot với tập dữ liệu là

$$A_i \Leftrightarrow Path_r(t) \Leftrightarrow Co_{r,i} \quad (3.3)$$

Khi số lượng robot tăng lên, một thách thức chính trong vận hành hệ thống là phối hợp tất cả các đơn vị để thực hiện các nhiệm vụ được yêu cầu một cách hiệu quả. Việc hệ thống nhận thức tình huống cũng rất quan trọng khi mỗi robot thực hiện quỹ đạo riêng của mình để xử lý tác vụ được giao trong không gian làm việc. Như được chỉ ra trong mối liên hệ (3.3), sự tương quan giữa các robot có giao cắt trong quỹ đạo mạnh hơn nhiều so với những robot không có giao cắt. Bên cạnh đó, vị trí giao cắt phù hợp hơn vị trí đích đến do phản ánh được quá trình di chuyển của robot, cũng phù hợp hơn vị trí hiện tại của robot, vốn biến đổi quá nhanh theo thời gian. Chúng có thể coi là những điểm nóng về giao thông của hệ thống đa robot trong không gian làm việc. Do đó, tập dữ liệu điểm giao cắt, được minh họa trong biểu thức (3.2), được chọn làm cơ sở cho phương pháp phân cụm robot. Điều này đặc biệt có lợi cho việc quản lý các đội robot di động một cách linh hoạt. Các robot phối hợp và chia sẻ nhận thức tình huống trong cùng một cụm thông qua một luồng dữ liệu chung,

hoạt động độc lập với các luồng dữ liệu của các cụm khác. Bảng 3.1 tổng hợp các ưu điểm và nhược điểm khi lựa chọn tập hợp điểm giao cắt là bộ dữ liệu đầu vào.

Bảng 3.1: Phân tích ưu điểm và nhược điểm của bộ dữ liệu

Ưu điểm	Tập trung vào vùng xung đột thực tế với việc phân cụm dựa trên các giao điểm giúp thuật toán tập trung tài nguyên tính toán vào những khu vực có mật độ xung đột cao nhất. Những khu vực robot di chuyển tự do, không có giao cắt sẽ không làm nhiều mô hình phân cụm.
	Giảm chiều dữ liệu và chi phí tính toán do số lượng giao điểm ít hơn nhiều so với việc lấy mẫu toàn bộ vị trí trên quỹ đạo; giúp thuật toán phân cụm hội tụ nhanh hơn, đáp ứng yêu cầu thời gian thực.
	Tính tiên đoán do dữ liệu này phản ánh rủi ro trong tương lai gần, nơi các robot sẽ gặp nhau, cho phép hệ thống phân cụm để quản lý trước khi tắc nghẽn thực sự xảy ra.
Nhược điểm	Bỏ qua va chạm song song khi chỉ xét điểm giao cắt; trường hợp robot di chuyển song song trong hành lang hẹp, nơi không có điểm cắt nhưng vẫn cần phối hợp để tránh va chạm.

Một đặc điểm khác của tập dữ liệu điểm giao cắt, xác định theo (3.2), là dữ liệu mang tính chất biến thiên theo thời gian. Trong thực tế vận hành, các phần tử robot liên tục được thiết lập các quỹ đạo mới căn cứ theo tác vụ được phân công. Do đó, bất kỳ sự thay đổi nào trong kế hoạch đường đi của một hoặc nhiều robot, xác định bởi MRPP, đều dẫn đến sự thay đổi vị trí và số lượng các điểm giao cắt. Tính chất động của phương pháp DCM chính là khả năng tự động tái cấu trúc các cụm theo sự thay đổi của dữ liệu đầu vào. Điều này cho phép cấu trúc nhóm luôn phản ánh chính xác nhất các khu vực xung đột tiềm năng hiện thời.

Thuật toán 4: Tìm ma trận điểm giao cắt

Đầu vào: Số lượng phần tử robot N ; Kích thước bản đồ map ; Mật độ vật cản obs_den

Đầu ra: Ma trận điểm giao cắt $intersections$

1 BẮT ĐẦU

2 Khởi tạo môi trường với N , map , và obs_den ; thiết lập tọa độ điểm xuất phát ($start_points$) và điểm đích đến ($goal_points$) theo hàm *random*.

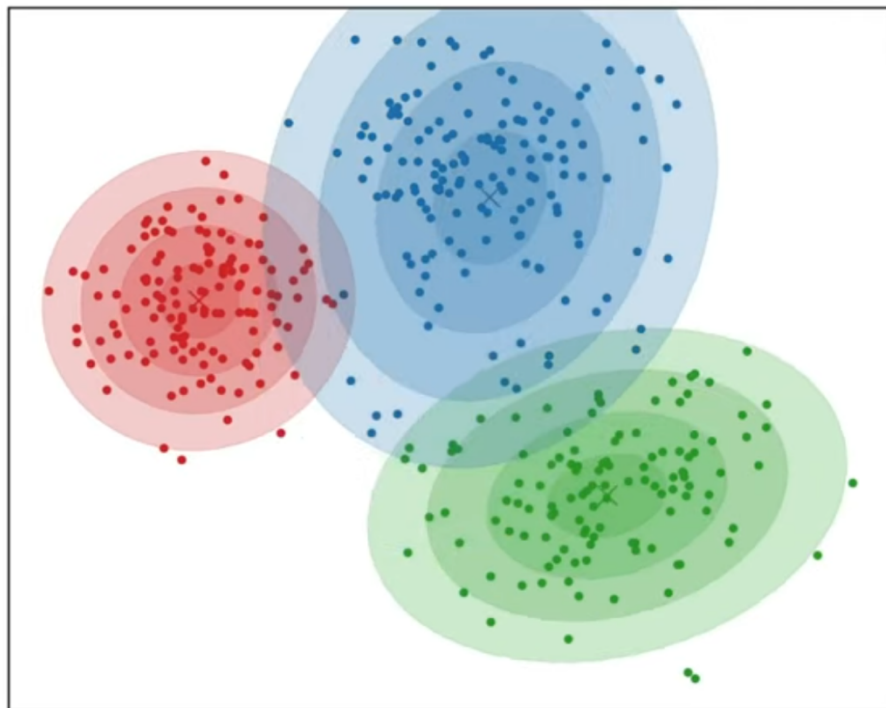
3 Thực thi thuật toán hoạch định đường đi

$$paths = A^*(start_points, goal_points) \text{ trong } map$$

4 Vòng lặp: trích xuất các điểm giao cắt của từng cặp đường đi

$$intersections(i,j) = intersect(paths(i), paths(j))$$

5 KẾT THÚC



Hình 3.2: Mô hình với ba cụm phân phối Gauss trong không gian ba chiều

3.3 Cơ sở toán học: mô hình GMM và thuật toán EM

Trong lý thuyết về thống kê trước đây và lĩnh vực học máy sau này, phân phối Gauss (phân bố chuẩn) là một trong những công cụ quan trọng nhất và có tính ứng dụng rất phổ biến. Mô hình phân phối Gauss hỗn hợp được hình thành từ tổ hợp tuyến tính của hai hay nhiều hàm phân bố chuẩn, được biết đến từ cuối thế kỷ 19. Tuy nhiên,

đến những năm 1990 - 2000 thông qua những công trình nghiên cứu của McLachlan và cộng sự, GMM đã được phổ biến rộng rãi trong những nghiên cứu và ứng dụng chuyên sâu về lĩnh vực học máy. Mô hình này đã mở rộng ưu điểm của hàm phân phối Gauss đơn lẻ, cho phép mô tả tập dữ liệu có mật độ phức tạp hơn. Sự kết hợp của nhiều hàm phân bố chuẩn và khả năng điều chỉnh một cách phù hợp các giá trị trung bình và hiệp phương sai cho phép mô hình có thể xấp xỉ bất kỳ mật độ liên tục nào với độ chính xác tùy ý. Hình 3.2 minh họa một ví dụ với ba hàm phân bố chuẩn trong không gian ba chiều và các biểu thức toán học dưới đây được mô tả dựa theo công bố của McLachlan và Rathnayake [149] cùng với sách "Mathematics for Machine Learning" [150].

Xét trường hợp tổng quát về một tổ hợp tuyến tính của K hàm mật độ xác suất tuân theo phân bố chuẩn, nó được biểu diễn thông qua công thức sau

$$p(\mathbf{x}) = \sum_{i=1}^K w_i \mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i) \quad (3.4)$$

trong đó $\mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i)$ được gọi là thành phần thứ i của mô hình hỗn hợp, với giá trị trung bình là μ_i và ma trận hiệp phương sai Σ_i ; phương trình mô tả cụ thể đối với thành phần $\mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i)$ là như sau

$$\mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i) = \frac{1}{(2\pi)^{\frac{n}{2}}(|\Sigma_i|)^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu_i)^T \Sigma_i^{-1}(\mathbf{x} - \mu_i)\right) \quad (3.5)$$

Trong công thức (3.4), với giả định hai phía của công thức đều là phân bố chuẩn, giá trị w_i được gọi là trọng số trộn của thành phần thứ i ; các giá trị w_i phải thỏa mãn điều kiện sau

$$\begin{cases} w_i > 0 \\ \sum_{i=1}^K w_i = 1 \end{cases} \quad (3.6)$$

Ba công thức (3.4), (3.5), và (3.6) là các phương trình toán học biểu diễn của mô hình hỗn hợp Gauss (GMM), trong đó các tham số của mô hình lần lượt là số

lượng thành phần K , giá trị trung bình là μ_i , ma trận hiệp phương sai Σ_i , và trọng số trộn w_i .

Một biến ngẫu nhiên nhị phân $\mathbf{z}$, còn gọi là biến ẩn, có chiều bằng K ; trong đó tồn tại một phần tử $z_i = 1$ và các giá trị còn lại $z_j = 0$, với $i \neq j$. Biến $\mathbf{z}$ thoả mãn điều kiện sau

$$\begin{cases} z_i \in \{0,1\} \\ \sum_{i=1}^K z_i = 1 \end{cases} \quad (3.7)$$

và là "nhãn" gán cho điểm dữ liệu tương ứng. Giá trị của biến $\mathbf{z}$ không thể tính toán được trực tiếp, chỉ có thể ước lượng qua giá trị xác suất trung gian. Khi đó, một hàm phân phối xác suất kết hợp $p(\mathbf{x}, \mathbf{z})$ được đề xuất bao gồm hai thành phần: phân phối biên $p(\mathbf{z})$ và phân phối điều kiện $p(\mathbf{x}|\mathbf{z})$, được biểu diễn theo phương trình sau

$$p(\mathbf{x}, \mathbf{z}) = p(\mathbf{z})p(\mathbf{x}|\mathbf{z}) \quad (3.8)$$

trong đó hàm phân phối biên được xác định theo trọng số trộn w_i thông qua công thức

$$p(z_i = 1) = w_i \quad (3.9)$$

Vì $\mathbf{z}$ biểu diễn tổ hợp K khả năng nên hàm phân phối biên (3.9) được viết lại theo dạng tổng quát như sau

$$p(\mathbf{z}) = \prod_{i=1}^K w_i^{z_i} \quad (3.10)$$

Tương tự, phân phối điều kiện của $\mathbf{x}$ với một giá trị cụ thể của $\mathbf{z}$ cũng là một phân bố chuẩn

$$p(\mathbf{x}|z_i = 1) = \mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i) \quad (3.11)$$

và được viết lại theo công thức tổng quát sau

$$p(\mathbf{x}|\mathbf{z}) = \prod_{i=1}^K \mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i)^{z_i} \quad (3.12)$$

Viết lại công thức (3.8) căn cứ theo (3.10) (3.12) để thể hiện hàm phân phối tổng quát của tất cả các khả năng có thể có của biến z thì thu được dạng biểu diễn của mô hình GMM ban đầu, tương tự như công thức (3.4)

$$p(\mathbf{x}) = \sum_{\mathbf{z}} p(\mathbf{z})p(\mathbf{x}|\mathbf{z}) = \sum_{i=1}^K w_i \mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i)$$

Dựa vào định lý Bayes, xác suất có điều kiện của z khi biết $\mathbf{x}$ là

$$\gamma(z_i) = p(z_i = 1|\mathbf{x}) = \frac{w_i \mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i)}{\sum_{j=1}^K w_j \mathcal{N}(\mathbf{x}|\mu_j, \Sigma_j)} \quad (3.13)$$

trong đó w_i là xác suất tiên nghiệm của $z_i = 1$, $\gamma(z_i)$ là xác suất hậu nghiệm sau khi đã quan sát dữ liệu $\mathbf{x}$; đồng thời $\gamma(z_i)$ cũng là giá trị "trách nhiệm" mà thành phần thứ i nhận để giải thích cho việc quan sát $\mathbf{x}$. Như vậy, công thức (3.13) cho biết mức độ "trách nhiệm" của từng cụm với giá trị liên tục nằm trong đoạn 0 đến 1, được cập nhật từ niềm tin ban đầu với xác suất tiên nghiệm w_i dựa trên quan sát dữ liệu $\mathbf{x}$.

Hàm logarit khả năng cho tập dữ liệu quan sát $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ được đề xuất

$$\ln p(\mathbf{X}|\mu, \Sigma, w) = \sum_{n=1}^N \ln \left\{ \sum_{j=1}^K w_j \mathcal{N}(\mathbf{x}|\mu_j, \Sigma_j) \right\} \quad (3.14)$$

Đây là hàm mục tiêu cần tối đa hoá để xác định bộ thông số tối ưu của GMM để mô hình hoá tập dữ liệu quan sát $\mathbf{X}$; khi đó, xác suất quan sát được bộ dữ liệu đạt lớn nhất. Thách thức lớn nhất của việc tìm kiếm lời giải là sự xuất hiện của các điểm dị biệt, khi một thành phần co lại và khớp chính xác một điểm dữ liệu ($\mu_j = \mathbf{x}_i$ và $\sigma_j \rightarrow 0$), hàm khả năng sẽ tiến tới vô cùng, dẫn đến bài toán không xác định và gây ra hiện tượng phù hợp quá mức. Một vấn đề khác là tính "định danh" xuất phát từ khả năng hoán đổi giữa các thành phần trong mô hình GMM; tồn tại tổ hợp K giải pháp tham số tương đương. Ngoài ra, việc tổng các thành phần nằm bên trong hàm logarit khiến cho các tham số của các thành phần Gauss khác nhau bị ràng buộc với nhau trong phương trình đạo hàm. Việc đặt đạo hàm bằng 0 không mang lại lời giải dạng đóng vì không thể giải riêng biệt từng tham số. Chính vì những khó khăn trong việc

tìm lời giải trực tiếp và nguy cơ từ các điểm dị biệt, thay vì sử dụng các phương pháp gradient thông thường, phương pháp dựa trên thuật toán Expectation-Maximization (EM) được sử dụng để tìm cực đại địa phương của hàm hợp lý một cách hiệu quả và ổn định hơn.

Cơ chế hoạt động của thuật toán EM được hình thành dựa vào liên hệ giữa hai biểu thức (3.13) và (3.14); trong đó (3.13) là kết quả của việc lấy đạo hàm biểu thức (3.14). EM cung cấp một cách tiếp cận lặp đi lặp lại để tìm các tham số của mô hình GMM thông qua việc tính toán giá trị "trách nhiệm" của mỗi thành phần đối với từng điểm dữ liệu, một cách gián tiếp ước lượng biến ẩn z . Quy trình gồm các bước sau:

Bước khởi tạo: chọn giá trị ban đầu ngẫu nhiên cho các tham số giá trị trung bình μ_i , ma trận hiệp phương sai Σ_i , và trọng số trộn w_i ; từ đó tính được giá trị khởi tạo cho hàm logarit khả năng.

Bước E, kỳ vọng: dựa trên tham số hiện tại, đánh giá các giá trị "trách nhiệm" theo biểu thức (3.13), được viết lại như sau

$$\gamma(z_i) = f_E(C_i|x_j) = \frac{w_i \mathcal{N}(x_j|\mu_i, \Sigma_i)}{\sum_{k=1}^K w_k \mathcal{N}(x_j|\mu_k, \Sigma_k)} \quad (3.15)$$

Bước M, tối đa: cập nhật lại các tham số dựa vào giá trị "trách nhiệm" $\gamma(z_i)$

$$\mu_i = \frac{\sum_{j=1}^N f_E(C_i|x_j) x_j}{N w_i} \quad (3.16)$$

$$\Sigma_i = \frac{\sum_{j=1}^N f_E(C_i|x_j) (x_j - \mu_i)(x_j - \mu_i)^T}{N w_i} \quad (3.17)$$

$$w_i = \frac{\sum_{j=1}^N f_E(C_i|x_j)}{N} \quad (3.18)$$

Bước lặp: tính toán giá trị của hàm logarit khả năng (3.14), nếu sự thay đổi nhỏ hơn ngưỡng hoặc tham số không đổi thì dừng lại; nếu không thì quay lại Bước E.

Thuật toán 5: Khởi tạo mô hình hỗn hợp Gauss và thuật toán EM

Đầu vào: Ma trận điểm giao cắt *intersections*

Đầu ra: Bộ thông số cho từng cụm của mô hình hỗn hợp Gauss: giá trị kỳ vọng (*means*), ma trận hiệp phương sai (*covs*), và trọng số trộn (*w_gmm*)

1 **BẮT ĐẦU**

2 Khởi tạo mô hình hỗn hợp GMM: số lượng cụm (*K_gmm*), số vòng lặp (*maxIterations*)

3 Khởi tạo bộ thông số cho từng cụm: giá trị kỳ vọng (*means*), ma trận hiệp phương sai (*covs*), và trọng số trộn (*w_gmm*)

4 **VÒNG LẶP:**

5 Tính toán trách nhiệm (bước E), xác định xác suất hậu nghiệm của từng điểm dữ liệu theo cụm

 * Biểu thức (3.15)

6 Cập nhật bộ thông số (bước M)

 * *means* theo biểu thức (3.16)

 * *covs* theo biểu thức (3.17)

 * *w_gmm* theo biểu thức (3.18)

7 Phân chia điểm dữ liệu thuộc *intersections* về các cụm

8 Tính toán giá trị hàm logarit khả năng (*log*) theo biểu thức (3.14)

9 NẾU $\log(i) - \log(i-1) < 1e-6$ THÌ thoát khỏi vòng lặp

10 NẾU vòng lặp $< \text{maxIterations}$ THÌ quay lại 4.

11 **KẾT THÚC**

3.4 Thiết lập hàm mục tiêu cân bằng tải

Để đánh giá về yếu tố cân bằng tải giữa các cụm, biểu thị qua sự phối hợp giữa các phần tử trong hệ thống đa robot, tham số "chi phí" hoàn toàn mới được đề xuất. Chi phí này được xác định bởi số lượng robot tham gia chia sẻ thông tin với nhau trong một khoảng thời gian xác định. Xét cho từng nhóm robot, dựa trên mối quan hệ tương tác cụ thể, thì giá trị chi phí của từng nhóm là

$$\delta_k = \frac{1}{N_K} \quad (3.19)$$

trong đó $k = 1..K$ với K là số lượng cụm và N_K là số lượng robot được phân bổ trong cụm thứ k .

Trong hệ thống như vậy, tham số δ_k đóng vai trò như là chỉ số đánh giá độ phức tạp quản lý của hệ thống, đại diện cho mức độ trung bình về tài nguyên truyền thông và xử lý phối hợp mà từng phần tử robot được nhận. Cụ thể hơn, khi số lượng phần tử càng lớn, giá trị tham số δ_k càng nhỏ, phản ánh mức độ chia sẻ thông tin thấp, độ trễ phối hợp tăng lên tương ứng. Ngược lại, khi số lượng phần tử càng nhỏ, giá trị tham số δ_k lớn, mức độ chia sẻ thông tin nhanh hơn, độ trễ phối hợp giảm xuống cải thiện khả năng nhận thức tình huống. Như vậy, việc phân cụm góp phần tác động trực tiếp vào giá trị chi phí, chia nhỏ bài toán lớn thành các bài toán nhỏ hơn để dễ quản lý.

Hơn nữa, trong hoạt động của thuật toán EM để xác định tập tham số cho các thành phần của mô hình GMM, bản chất lặp của thuật toán EM tạo ra nhiều kết quả tiềm năng khác nhau. Hàm mục tiêu được đề xuất để đạt được kết quả phân cụm gần tối ưu, cụ thể là

$$f_{obj} = \sqrt{\sum_{k=1}^K (\delta_k)^2} \rightarrow \min \quad (3.20)$$

Đây là hàm mục tiêu bình phương tối thiểu, xác định dựa theo tham số chi phí cho từng thành phần của mô hình GMM; vai trò như một cơ chế lọc để buộc hệ thống hướng đến sự cân bằng tải thông qua lựa chọn được cấu hình phân cụm phù hợp nhất. Nó cho phép loại bỏ những trường hợp phân cụm thiên lệch, tạo ra cụm có số lượng phần tử quá nhỏ, gây lãng phí tài nguyên; ưu tiên lựa chọn những phương án có số lượng robot được phân phối đồng đều nhất giữa các cụm đã xác định. Khi đó mối quan hệ giữa hàm mục tiêu f_{obj} và tham số chi phí δ_k là mong muốn tạo ra các nhóm nhỏ được cân đối cùng với khả năng tạo cân bằng giữa các nhóm.

Xem xét theo tính chất động của phương pháp DCM cũng như đặc điểm của tập dữ liệu điểm giao cắt, hàm mục tiêu f_{obj} đóng vai trò như một tiêu chuẩn ổn định. Khi các điểm giao cắt thay đổi do cập nhật đường đi, hệ thống có thể có thêm nhiều phương án tái phân cụm khác nhau. Việc duy trì điều kiện (3.20) trong suốt quá trình vận hành sẽ đảm bảo rằng cấu trúc cụm có thay đổi linh hoạt để thích ứng với các điểm giao cắt mới thì hệ thống vẫn luôn duy trì được trạng thái cân bằng tải tốt nhất có thể. Bảng 3.2 mô tả phân tích chi tiết hơn về hàm mục tiêu với tham số chi phí, xét trong bối cảnh quan hệ với dữ liệu điểm giao cắt. Nói cách khác, thuật toán DCM sử dụng hàm mục tiêu này để định vị cấu trúc hệ thống vào trạng thái tối ưu về mặt quản lý, ngay cả khi các yếu tố hình học (vị trí giao cắt) có biến động.

Bảng 3.2: Phân tích mối quan hệ giữa phối hợp và cân bằng

Yếu tố	Động lực từ dữ liệu giao cắt	Động lực từ hàm mục tiêu và tham số chi phí	Kết quả mong đợi
Cấu trúc cụm	Muốn gom các robot có khả năng va chạm vào một nhóm; có thể tạo ra các nhóm rất to hoặc rất nhỏ tùy vào mật độ giao thông.	Muốn các nhóm có kích thước đồng đều nhau; tránh việc tạo ra một nhóm quá đông hoặc nhóm chỉ có một vài phần tử.	Cụm tối ưu cục bộ: các robot có nguy cơ va chạm được gom lại, kích thước nhóm được kiểm soát để không quá chênh lệch.
Phối hợp	Đảm bảo an toàn; robot va chạm buộc phải chia sẻ thông tin với nhau.	Đảm bảo hiệu suất, tối ưu hóa băng thông bằng cách không để nhóm nào quá tải.	Hiệu quả hệ thống: robot nhận thức được tình huống va chạm nhưng không bị quá tải thông tin.

Thuật toán 6: Thuật toán phân cụm động DCM

1 BẮT ĐẦU

- 2 Gọi Thuật toán 4: Tìm ma trận điểm giao cắt
 - 3 Khởi tạo: số vòng lặp ($maxIterations$)
 - 4 VÒNG LẶP:
 - 5 Gọi Thuật toán 5: Khởi tạo mô hình hỗn hợp Gauss và thuật toán EM
 - 6 Phân chia phần tử robot về các cụm, tính số lượng robot thuộc về mỗi cụm ($compCounts$)
 - 7 Tính giá trị hàm mục tiêu theo biểu thức (3.20), cập nhật giá trị obj_min (nếu có giá trị tốt hơn).
 - 8 NẾU giá trị hàm mục tiêu < giá trị obj_min THÌ cập nhật giá trị obj_min mới VÀ cập nhật cấu hình phân cụm tốt nhất.
 - 9 NẾU vòng lặp < $maxIterations$ THÌ quay lại 4.
 - 10 Xuất cấu hình phân cụm tốt nhất.
 - 11 KẾT THÚC
-

3.5 Quy trình thực hiện DCM

Phương pháp DCM được đề xuất bao gồm hai giai đoạn chính với đầu vào là tập dữ liệu giao cắt. Cụ thể:

i. Giai đoạn 1: Mô hình GMM được thiết lập với đầu vào là tập dữ liệu điểm giao cắt, sử dụng phương trình (3.4) và (3.5). Xác định các bộ tham số cho mô hình GMM dựa trên thuật toán EM, bao gồm: giá trị trung bình μ_i , ma trận hiệp phương sai Σ_i , và trọng số w_i cho mỗi cụm. Thuật toán lặp EM được triển khai tuần tự theo bốn bước: khởi tạo, bước kỳ vọng E với biểu thức (3.15), bước tối đa M với các biểu thức (3.16), (3.17), và (3.18), và bước kiểm tra điều kiện lặp với ngưỡng so sánh.

ii. Giai đoạn 2: Xác định số lượng robot trong mỗi cụm dựa trên quan hệ giữa mỗi phần tử robot với tập dữ liệu thông qua biểu thức (3.3). Trên cơ sở đó, tính toán giá trị chi phí của từng cụm và hàm mục tiêu theo biểu thức (3.19) và (3.20).

Giai đoạn 1 và Giai đoạn 2 được thực hiện lặp lại để tìm giá trị nhỏ nhất của hàm mục tiêu sau một số vòng lặp nhất định. Bộ tham số tương ứng cho mô hình GMM khi đó cũng chính là kết quả gần tối ưu cho yêu cầu cân bằng tải giữa các cụm.

3.6 Kết luận chương 3

Nội dung của chương này đã trình bày chi tiết về quá trình xây dựng phương pháp phân cụm động, được sử dụng để chia các phần tử trong hệ thống đa robot vào những cụm nhỏ hơn; có hai đặc điểm quan trọng. Thứ nhất là lựa chọn cách tiếp cận phân cụm mềm, DCM tích hợp GMM cùng với thuật toán EM, cho phép mô hình hóa sự phân bố robot theo xác suất. Thứ hai là đề xuất tham số chi phí δ_k đóng vai trò như là chỉ số đánh giá độ phức tạp quản lý của hệ thống và hàm mục tiêu toàn phương f_{obj} có vai trò là bộ lọc để lựa chọn được cấu hình phân cụm cân bằng tải: các cụm có số lượng phần tử cân bằng với nhau.

Những công bố liên quan đến Chương 3 có: Công trình 1, 3, 5, và 6 trong Danh mục các công trình khoa học.

Chương 4. THỬ NGHIỆM SỐ VÀ ĐÁNH GIÁ HOẠT ĐỘNG

4.1 Đặt vấn đề

Dựa trên, phương trình toán học, quy trình thực hiện và khung thuật toán cơ bản đã được xây dựng ở Chương 2 (đối với ElasticABC) và Chương 3 (đối với DCM), nghiên cứu sinh đã phát triển các chương trình mô phỏng trên máy tính sử dụng phần mềm MATLAB, để thử nghiệm và đánh giá hoạt động của ElasticABC và DCM. Nội dung chương này trình bày chi tiết về quá trình thử nghiệm và đánh giá hoạt động. Khi thực thi các chương trình mô phỏng trong nhiều kịch bản thử nghiệm khác nhau, kết quả thu được là cơ sở phân tích và đánh giá hoạt động. Hơn nữa, việc so sánh hoạt động của ElasticABC với thuật toán ABC truyền thống được thực hiện để góp phần làm rõ hơn điểm cải thiện vượt trội cũng như sự khác biệt của thuật toán mới.

Đối tượng thử nghiệm là hệ thống đa robot có kiến trúc điều khiển tập trung, các phần tử robot có cấu tạo tương tự nhau, hoạt động theo sự quản lý, giám sát từ một trạm điều khiển trung tâm. Khi mỗi phần tử được phân công một tác vụ, tương ứng với vị trí bắt đầu và vị trí đích đến cụ thể, thuật toán ElasticABC sẽ thực thi để tìm kiếm đường đi gần tối ưu cho từng robot. Khi mở rộng bài toán, bên cạnh ElasticABC, những thuật toán khác cũng có thể được dùng để hoạch định đường đi, ví dụ như A* hoặc ABC truyền thống. Trong quá trình robot di chuyển theo hành trình, thực thi tác vụ, dữ liệu hoạt động và vận hành được từng phần tử chia sẻ về trạm điều khiển trung tâm; ngược lại trạm điều khiển trung tâm cung cấp dữ liệu vận hành của hệ thống đến các phần tử robot. Thuật toán DCM được dùng để phân chia các phần tử thành cụm nhỏ hơn, khi đó dữ liệu vận hành của hệ thống được gửi theo cụm. Thử nghiệm được tiến hành với kịch bản: (1) đối với thuật toán ABC truyền thống, hệ thống có 3 phần tử; (2) đối với thuật toán ElasticABC, hệ thống có 10 phần tử; (2) đối với thuật toán DCM, hệ thống có 50 phần tử và xem xét trường hợp thu gọn quy mô chỉ còn 25 phần tử. Việc lựa chọn giá trị cụ thể không giảm tính tổng quát của thuật toán do tham số đã được quy định một biến đầu vào.

4.2 Đánh giá quần thể bầy ong

Trong quá trình xây dựng thuật toán ElasticABC, nghiên cứu sinh đã tiến hành đánh giá quần thể bầy ong theo mô hình phi tập trung và tập trung cho thuật toán ABC truyền thống khi áp dụng vào bài toán MRPP. Trong mô hình thứ nhất, đường đi của mỗi robot được tìm kiếm độc lập, không cần quan tâm đến trạng thái của các robot khác; sử dụng hàm mục tiêu riêng với hai thành phần: độ dài đường đi và hàm phạt tránh vật cản tĩnh. Mô hình thứ hai tìm kiếm tập hợp tuyến đường tối ưu chung và đồng thời cho tất cả phần tử robot; sử dụng hàm mục tiêu chung gồm ba thành phần: tổng độ dài đường đi, hàm phạt va chạm với vật cản tĩnh và hàm phạt va chạm giữa các robot với nhau. Khi thử nghiệm, một điểm cải tiến đã được áp dụng cho mô hình thứ hai là khởi tạo chọn lọc thay thế cho khởi tạo ngẫu nhiên; cụ thể lấy kết quả thử nghiệm của mô hình thứ nhất làm giải pháp khởi tạo của mô hình thứ hai.

Cụ thể hơn, hàm mục tiêu của mô hình thứ nhất xác định theo công thức (2.11), hàm thích nghi là công thức (2.16). Với mô hình thứ hai, hàm mục tiêu là

$$f_{obj} = f_{obj,L} + f_{obj,O} + f_{obj,C} \quad (4.1)$$

trong đó $f_{obj,L}$ là tổng độ dài đường đi của các phần tử robot, được xác định theo phương trình sau

$$f_{obj,L} = \sum_{i=1}^N L_i = \sum_{i=1}^N \left(\sum_{j=1}^{m-1} \|p_{i,j+1} - p_{i,j}\|_2 \right) \quad (4.2)$$

$f_{obj,O}$ và $f_{obj,C}$ lần lượt là phạt va chạm với vật cản tĩnh và phạt va chạm giữa các robot, được xác định như sau

$$f_{obj,O} = \omega_1 \sum_{j=1}^m S(i, j) \quad (4.3)$$

$$f_{obj,C} = \omega_2 \sum_{i=1}^{N-1} \sum_{k=i+1}^N \sum_{j=1}^m C(i, j, k) \quad (4.4)$$

trong đó ω_1 và ω_2 là các hệ số phạt, có giá trị rất lớn; $C(i, j, k)$ là hàm va chạm

$$C(i, j, k) = \begin{cases} 1, & \text{if } \|p_{i,j+1} - p_{i,j}\|_2 < d_{safe} \\ 0, & \text{otherwise} \end{cases} \quad (4.5)$$

Khi đó, hàm thích nghi của mô hình tập trung sẽ được xác định như sau

$$fit = \frac{1}{1 + f_{obj}} \quad (4.6)$$

với f_{obj} được tính toán từ phương trình (4.1).

Thử nghiệm được thực hiện với nhóm nhỏ 03 phần tử robot, tập hợp điểm bắt đầu và điểm đích đến được giữ cố định; kích thước bản đồ là 50x50, vị trí vật cản được bố trí ngẫu nhiên, mật độ vật cản 20% tổng diện tích. Thông số thiết lập cho thuật toán ABC được mô tả trong Bảng 4.1.

Bảng 4.1: Thông số của thuật toán ABC

Thông số	Giá trị	Mô tả
Kích thước quần thể	25	Tổng số lượng ong trong bầy.
Số lượng điểm trung gian	5	Số lượng điểm trung gian có thể chỉnh sửa thành đường đi giữa điểm bắt đầu và điểm kết thúc.
Ngưỡng từ bỏ	50	Số lần thử liên tiếp mà ong thợ và ong quan sát thực hiện trước khi từ bỏ một giải pháp.
Số vòng lặp tối đa	200	Số chu kỳ tối đa mà thuật toán thực hiện.

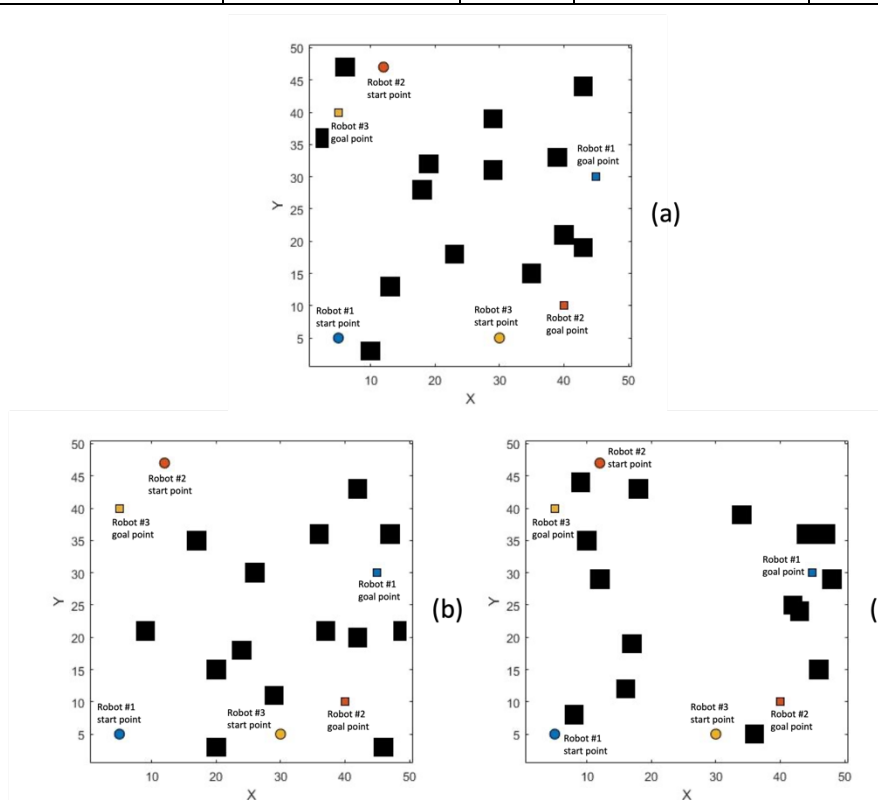
Thuật toán ABC được chạy lặp lại 50 lần với hai mô hình quần thể ở cả ba phương án bố trí vật cản, được lựa chọn ngẫu nhiên. Trong phương án thứ nhất, mô hình quần thể phi tập trung có chi phí trung bình xấp xỉ 137,44 cùng độ lệch chuẩn ở mức 0,57 và hệ số CV chỉ là 0,41%. Mô hình quần thể tập trung có chi phí trung bình lớn hơn (142,72) cùng với độ lệch chuẩn cao hơn (7,32) và hệ số CV là 5,1%. Các giá trị của chỉ số thông kê được tính toán từ giá trị chi phí của cả 50 lượt chạy, minh họa trong Bảng 4.2. Hệ số CV rất thấp, chỉ 0,41%, cho thấy dữ liệu mô phỏng của mô hình phi tập trung có mức biến động rất thấp, thể hiện độ tin cậy và lặp lại của kết quả. Trong khi biến động trong dữ liệu của mô hình tập trung lớn hơn (hệ số CV lên

tới 5,1%), xác nhận sự tồn tại của điểm dữ liệu ngoại lai, phản ánh sự khó khăn của việc tìm kiếm đường đi khả thi. Tuy nhiên, mô hình tập trung lại đạt mức chi phí thấp hơn mô hình phi tập trung ở 18/50 lượt thử nghiệm (tỷ lệ đạt 36%); phản ánh hiệu quả của giải pháp khởi tạo chọn lọc.

Bảng 4.2: Giá trị chi phí của phương án thứ nhất

Lượt chạy	Quản thể phi tập trung	Quản thể tập trung	Lượt chạy	Quản thể phi tập trung	Quản thể tập trung
1	137,88	137,60	26	137,91	137,71
2	137,37	147,44	27	137,00	137,58
3	136,98	157,81	28	136,98	136,89
4	137,08	137,08	29	137,29	137,27
5	138,21	138,10	30	137,82	137,38
6	137,64	137,32	31	136,94	136,94
7	138,48	138,08	32	138,58	143,83
8	137,18	137,07	33	137,45	146,07
9	137,70	137,70	34	138,31	155,85
10	136,92	136,92	35	137,30	149,52
11	137,31	140,61	36	137,07	137,02
12	137,30	138,87	37	137,73	144,87
13	139,06	152,14	38	137,42	137,35
14	136,75	144,98	39	136,97	136,97
15	138,34	137,97	40	136,78	144,28
16	137,32	153,86	41	137,69	138,95
17	137,63	147,01	42	137,70	137,16
18	138,59	138,58	43	136,97	155,21

Lượt chạy	Quần thể phi tập trung	Quần thể tập trung	Lượt chạy	Quần thể phi tập trung	Quần thể tập trung
19	137,13	154,05	44	136,81	136,81
20	137,44	157,44	45	136,81	136,80
21	138,48	141,84	46	136,99	143,77
22	136,87	164,03	47	137,00	136,98
23	136,94	136,92	48	137,00	146,74
24	137,86	156,09	49	137,19	140,67
25	137,08	137,05	50	136,80	136,80



Hình 4.1: Bản đồ với phương án vị trí vật cản khác nhau

Ba phương án bố trí vật cản được minh họa như Hình 4.1, phương án thứ nhất là với Hình 4.1.a; phương án thứ hai và thứ ba lần lượt tương ứng với Hình 4.1.b và Hình 4.1.c được sử dụng để đánh giá về mức độ thích nghi của hai mô hình quần thể trong điều kiện môi trường thay đổi. Kết quả tính toán chỉ số thống kê: giá trị trung

binh, độ lệch chuẩn và hệ số biến thiên của cả hai phương án được minh họa trong Bảng 4.3. Số liệu cho thấy quần thể phi tập trung tiếp tục duy trì ổn định rất cao ở cả hai trường hợp khi mà hệ số CV luôn duy trì ở mức rất thấp, quanh mức 0,5%. Ngược lại, số liệu của quần thể tập trung vẫn duy trì mức biến động lớn, quanh mức 5%. Hơn thế nữa, mô hình quần thể tập trung phụ thuộc rõ rệt vào cấu hình vật cản; trong phương án thứ hai, quần thể tập trung có số lượt thử đạt mức chi phí thấp hơn quần thể phi tập trung gia tăng, ở mức 25/50 lượt thử nghiệm (tỷ lệ lên tới 50%). Ngược lại, số lượt thử đạt mức chi phí thấp hơn quần thể phi tập trung giảm mạnh, ở mức 13/50 lượt thử nghiệm (tỷ lệ chỉ còn 26%), phản ánh thách thức lớn về xung đột trong phương án này.

Bảng 4.3: Số liệu thống kê của phương án 2 và 3

Thông số tính toán	Quần thể phi tập trung	Quần thể tập trung	Thông số tính toán	Quần thể phi tập trung	Quần thể tập trung
Giá trị trung bình (μ)	137,82	141,55	Giá trị trung bình (μ)	138,24	142,13
Độ lệch chuẩn (σ)	0,69	8,86	Độ lệch chuẩn (σ)	0,61	7,13
Hệ số biến thiên (CV)	0,50%	6,26%	Hệ số biến thiên (CV)	0,44%	5,01%

Như vậy, hiệu quả tìm đường của quần thể tập trung sử dụng thuật toán ABC truyền thống phụ thuộc lớn vào cấu hình môi trường khi mà sự thay đổi vị trí vật cản ảnh hưởng trực tiếp đến độ phức tạp của việc phối hợp. Mặc dù mô hình quần thể tập trung đã áp dụng giải pháp khởi tạo chọn lọc ở giai đoạn khởi tạo quần thể, thể hiện mức độ cải thiện chi phí trong một số lượt thử nghiệm. Khi số lượng phần tử robot tăng lên, thách thức tìm kiếm càng lớn do sự bùng nổ không gian tìm kiếm. Các thử nghiệm tiếp theo cho thuật toán ElasticABC cũng được triển khai theo mô hình quần thể phi tập trung và có khả năng tự thích nghi cấu trúc cho từng phần tử robot.

4.3 Thử nghiệm và đánh giá hoạt động của thuật toán ElasticABC

Cải tiến cốt lõi của ElasticABC là tối ưu hoá cấu trúc, số lượng điểm trung gian được điều chỉnh thông qua cơ chế co giãn với cặp toán tử đối ngẫu hoạt động đồng thời trong giai đoạn tìm kiếm của ong thợ. Hình thức biểu diễn linh hoạt này cho phép tạo ra các đường đi tự thích nghi theo đặc điểm thực tế của không gian làm việc và quỹ đạo di chuyển cần thiết lập cho từng phần tử robot. Trên cơ sở đó, kịch bản thử nghiệm được thiết lập cho những điều kiện môi trường biến đổi để đánh giá định lượng hoạt động của ElasticABC, cung cấp minh chứng cụ thể và rõ ràng. Bảng 4.4 liệt kê thông số về môi trường thử nghiệm; Bảng 4.5 liệt kê các thông số được thiết lập cho thuật toán ElasticABC.

Bảng 4.4: Thông số môi trường

Thông số	Giá trị	Mô tả
N_ROBOTS	10	Số lượng robot trong hệ thống
MAP_SIZE	50	Kích thước bản đồ (50x50)
OBS_DENSITY	[15 20]	Mật độ vật cản tĩnh (%), tính theo diện tích chiếm chỗ trong bản đồ
OBS_DIM	[5 15]	Kích thước vật cản tĩnh (%)

Bảng 4.5: Thông số cho thuật toán ElasticABC

Thông số	Giá trị	Mô tả
nPop	25	Số lượng cá thể trong quần thể ong.
Limit	15	Số lần thử liên tiếp mà ong thợ và ong quan sát thực hiện trước khi từ bỏ một giải pháp
MaxIt	100	Số vòng lặp tối đa của giải thuật ElasticABC
SAFETY_MARGIN	0.5	Vùng đệm an toàn bao quanh vật cản

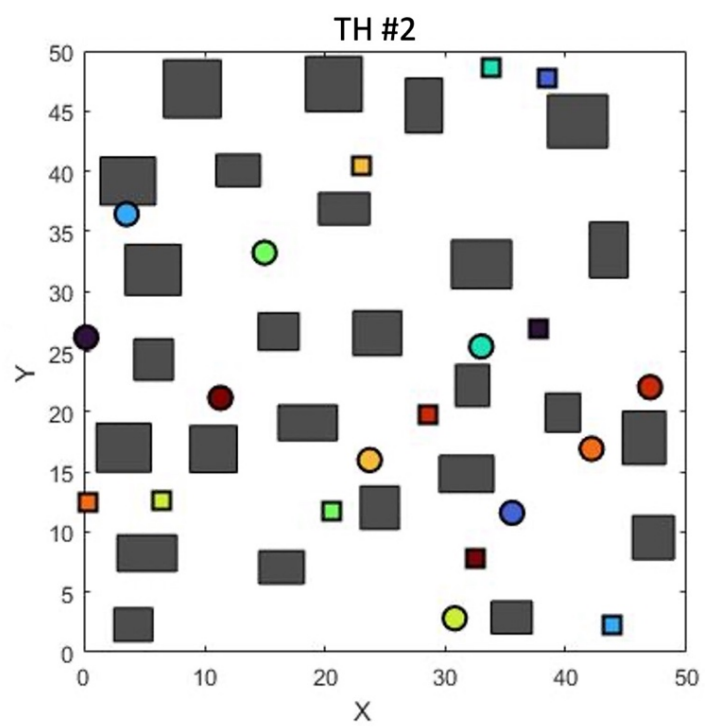
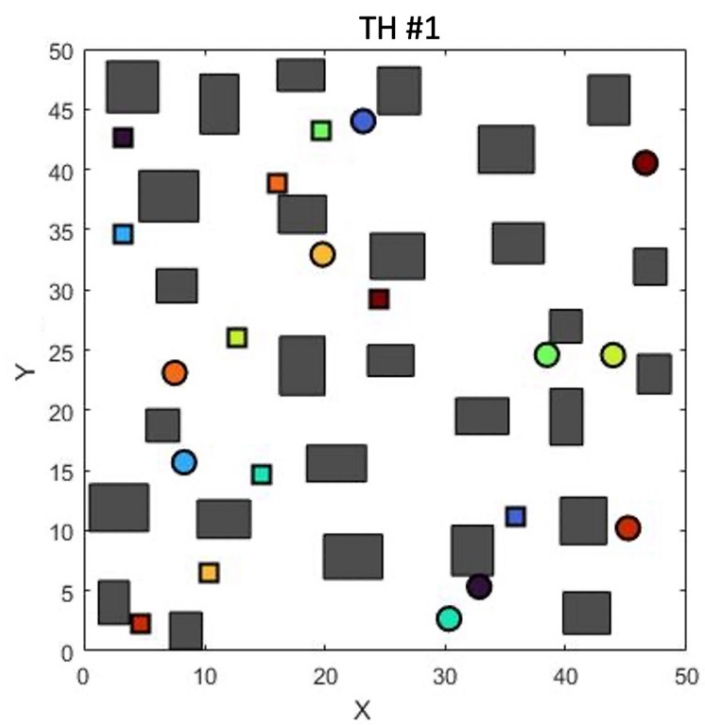
Thông số	Giá trị	Mô tả
d_safe	1.0	Khoảng cách an toàn cho công thức của toán tử Dẫn
epsilon	[0 1.0]	Hệ số ngẫu nhiên theo phương pháp tuyến trong công thức của toán tử Dẫn
lamda	[-1.0 1.0]	Hệ số ngẫu nhiên theo phương tiếp tuyến trong công thức của toán tử Dẫn
cosTheta	0.995	Hệ số ngưỡng của toán tử Co

Trong khi số lượng phần tử robot trong hệ thống và kích thước bản đồ được giữ cố định, mật độ vật cản tĩnh trong bản đồ và kích thước vật tĩnh được thay đổi theo kịch bản thử nghiệm; loại vật cản tĩnh với biên dạng lồi. Cụ thể:

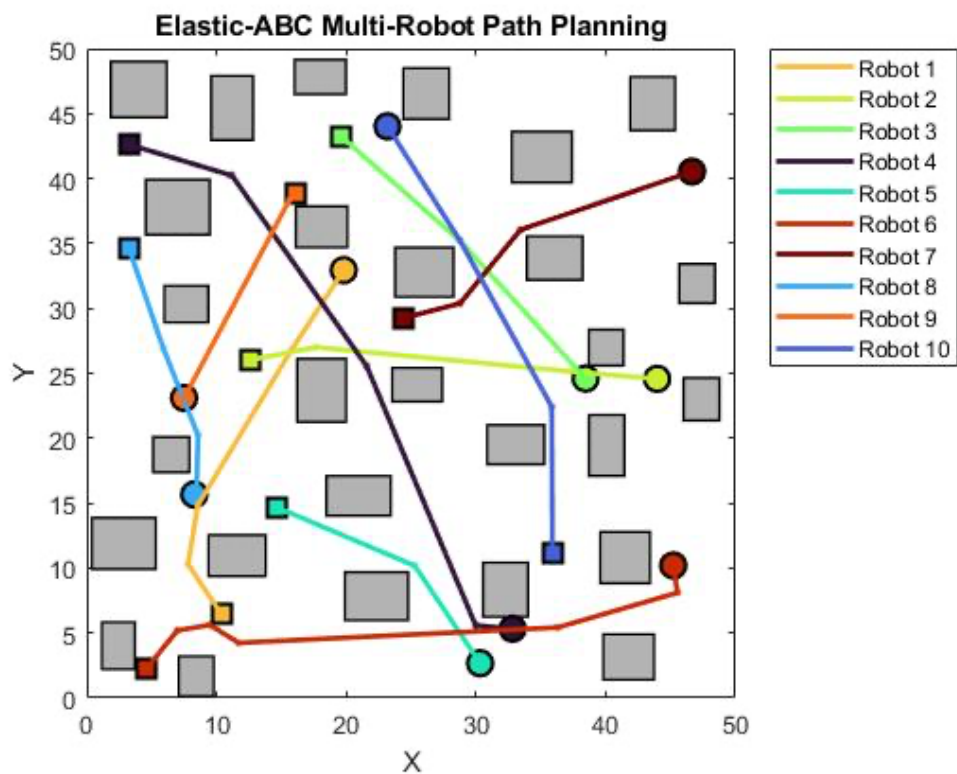
i. Trường hợp thứ nhất và thứ hai (TH #1 và TH #2): mật độ vật cản tĩnh là 15%, kích thước vật cản tĩnh trong khoảng 5% đến 10%; tương ứng với hai cấu hình vị trí bố trí vật cản khác nhau trong bản đồ, minh hoạ như Hình 4.2.

ii. Trường hợp thứ ba (TH #3): mật độ vật cản tĩnh là 20%; kích thước vật cản tĩnh trong khoảng 10% đến 15%.

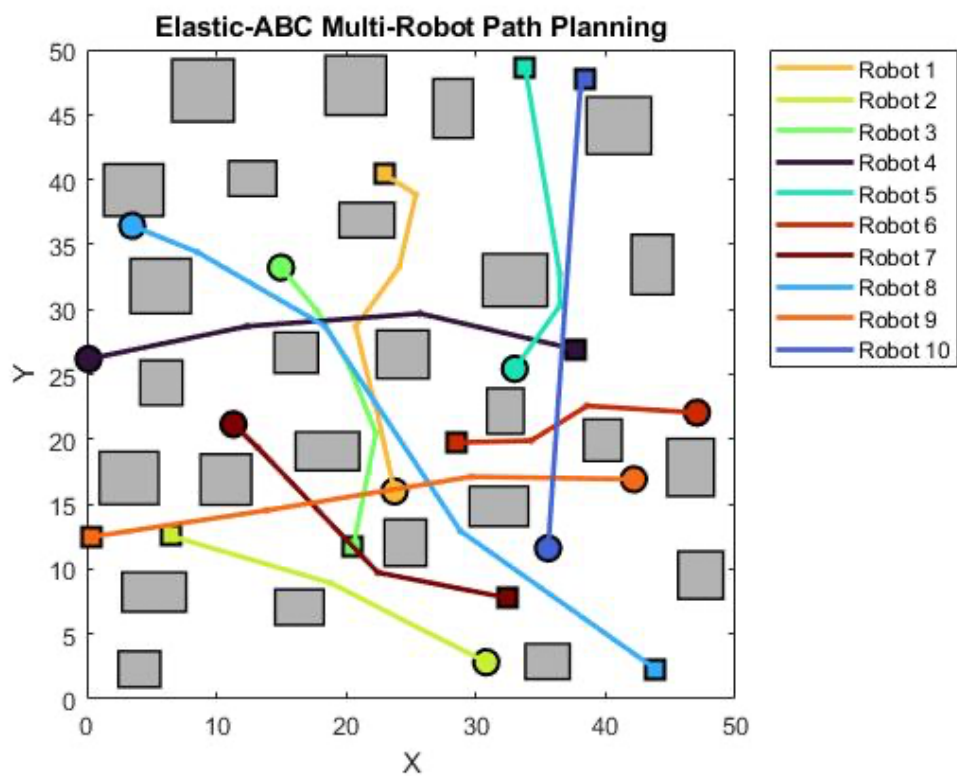
Hai cấu hình vật cản TH #1 và TH #2 được minh hoạ trong Hình 4.2; vị trí bắt đầu và vị trí đích đến của 10 phần tử robot cũng được khởi tạo ở các điểm ngẫu nhiên trong vùng không gian trống của bản đồ. Hình 4.3 minh hoạ đồ thị đường đi, là kết quả hoạch định của thuật toán ElasticABC đối với trường hợp TH #1, hình tròn là điểm bắt đầu, hình vuông là điểm kết thúc, mỗi màu biểu thị hành trình đã tìm được của một phần tử robot; Bảng 4.6 tổng hợp giá trị chi phí và số lượng điểm trung gian (WP) ở thời điểm hoàn thành khởi tạo phỏng đoán và thời điểm giai đoạn tìm kiếm kết thúc cho từng phần tử robot; mức giảm chi phí (%) và số lượng WP giữa hai thời điểm. Giá trị chi phí là kết quả cuối cùng của hàm mục tiêu, tương ứng với độ dài hành trình của từng robot. Tương tự với trường hợp TH #2, Hình 4.4 và Bảng 4.7 cung cấp kết quả về đồ thị và dữ liệu.



Hình 4.2: Kịch bản thử nghiệm TH #1 và TH #2



Hình 4.3: Kết quả đường đi tối ưu cho TH #1



Hình 4.4: Kết quả đường đi tối ưu cho TH #2

Bảng 4.6: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A1

Thứ tự robot	Khởi tạo phỏng đoán		ElasticABC		Mức giảm	
	Chi phí	WP	Chi phí	WP	% chi phí	WP
R1	35,400	10	30,501	5	13,84	5
R2	32,337	5	31,555	4	2,42	1
R3	26,562	3	26,536	3	0,10	-
R4	54,594	12	50,833	6	6,89	6
R5	21,419	5	20,531	3	4,15	2
R6	54,210	23	45,225	7	16,57	16
R7	27,178	8	25,789	4	5,11	4
R8	21,620	5	19,946	5	7,74	-
R9	17,972	3	17,895	3	0,43	-
R10	40,935	10	36,307	4	11,31	6
Tổng	332,227		305,118			

Bảng 4.7: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A2

Thứ tự robot	Khởi tạo phỏng đoán		ElasticABC		Mức giảm	
	Chi phí	WP	Chi phí	WP	% chi phí	WP
R1	33,786	17	27,661	7	18,13	10
R2	26,367	3	26,313	3	0,20	-
R3	28,335	9	23,771	5	5,70	4

Thứ tự robot	Khởi tạo phỏng đoán		ElasticABC		Mức giảm	
	Chi phí	WP	Chi phí	WP	% chi phí	WP
R4	40,529	7	38,219	4	27,02	3
R5	33,580	13	24,508	4	11,61	9
R6	21,828	6	19,294	4	6,39	2
R7	27,968	9	26,180	4	2,25	5
R8	55,401	8	54,154	5	2,26	3
R9	43,188	6	42,210	4	1,09	2
R10	36,773	3	36,373	3	8,36	-
Tổng	347,755		318,683			

Với Bộ dữ liệu #A1 trong Bảng 4.6: mức giảm lớn nhất ở robot R6 (16 điểm); R4, R7, và R10 có mức giảm trung bình (4 đến 6 điểm); R3, R8, và R9 không thay đổi. Với Bộ dữ liệu #A2 trong Bảng 4.7: mức giảm lớn nhất ở robot R5 và R10 (hơn 9 điểm); R3, R4, R7, và R8 có mức giảm trung bình (3 đến 5 điểm); R2 và R10 không thay đổi. Xu hướng cắt giảm điểm trung gian bộc lộ cho 10 phần tử; minh chứng rõ ràng cho hoạt động của cặp toán tử đối ngẫu đồng thời trong giai đoạn tìm kiếm của ong thợ: điểm trung gian điều chỉnh theo từng phần tử robot và theo điều kiện môi trường. Tỷ lệ giảm điểm trung gian cho toàn hệ thống vào khoảng 47% cho cả hai trường hợp.

Tương tự, xu hướng cải thiện chi phí rõ ràng cho cả 10 phần tử robot khi thuật toán hoạt động. Đặc biệt là sự tương quan giữa việc giảm nhiều số lượng WP ở những phần tử robot tiết kiệm được nhiều chi phí nhất; đạt mức giảm chi phí hơn 10%: R1, R6, và R10 với Bộ dữ liệu #A1 trong Bảng 4.6, R1, R4, và R5 với Bộ dữ liệu #A2 trong Bảng 4.7. Đây là minh chứng bổ sung cho việc điều chỉnh điểm trung gian của cặp toán tử đối ngẫu trong giai đoạn tìm kiếm của ong thợ. R8 (Bộ dữ liệu #A1) và

R10 (Bộ dữ liệu #A2) là hai trường hợp riêng, có mức giảm chi phí hơn 7% trong khi số lượng WP không đổi, phản ánh năng lực tìm kiếm giải pháp tốt hơn về chất lượng trong giai đoạn tìm kiếm ong thợ. Một số kết quả có mức giảm chi phí rất nhỏ, cho thấy đặc điểm liên quan đến khởi tạo phỏng đoán, tích hợp trong ElasticABC; giải pháp ban đầu có chất lượng trong khi vẫn đảm bảo được đặc tính ngẫu nhiên cần thiết; thuật toán đã tìm ra đường đi "tối ưu" ở giai đoạn tìm kiếm. Tỷ lệ giảm chi phí cho toàn hệ thống vào khoảng 8% cho cả hai trường hợp TH #1 và TH #2.

Kết luận ban đầu cho hoạt động của thuật toán ElasticABC là: (1) khả năng loại bỏ điểm trung gian dư thừa đồng thời với việc giảm chi phí; (2) khả năng thích nghi với dữ liệu khởi tạo chưa tốt: số lượng điểm trung gian lớn ở giai đoạn khởi tạo phỏng đoán nhưng đã giảm mạnh sau khi kết thúc giai đoạn tìm kiếm; (3) độ tin cậy luôn duy trì trong tất cả kết quả.

Để loại bỏ được yếu tố chủ quan của một dữ liệu đơn lẻ, hai trường hợp TH #1 và TH #2 được chạy lặp lại 50 lần và tính toán chỉ số thống kê cho tập dữ liệu hoạt động lớn hơn. Việc đánh giá về đặc điểm và quy luật hoạt động của thuật toán được chi tiết và cụ thể hơn so với một dữ liệu đơn lẻ. Bảng 4.8, Bảng 4.9, Bảng 4.10 và Bảng 4.11 minh họa dữ liệu 50 lượt chạy cho trường hợp TH #1 của tham số tương ứng: giá trị chi phí và số lượng điểm trung gian ở thời điểm hoàn thành giai đoạn khởi tạo phỏng đoán, giá trị chi phí và số lượng điểm trung gian ở thời điểm kết thúc giai đoạn tìm kiếm. Trường hợp TH #2 cũng có bộ dữ liệu kết quả mô phỏng tương tự.

Bảng 4.8: Giá trị chi phí sau giai đoạn khởi tạo phỏng đoán

Lượt chạy	Khởi tạo phỏng đoán/Chi phí									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
1	35,821	34,410	26,553	54,278	21,173	48,534	26,960	22,621	18,017	39,853
2	40,395	32,661	26,561	55,007	21,183	45,537	27,058	22,425	17,994	40,021
3	35,750	33,284	26,549	54,059	21,246	47,454	27,079	22,439	17,987	38,915
4	40,340	32,841	26,597	53,892	21,279	50,798	27,327	21,548	18,007	39,660
5	41,681	32,275	26,565	54,856	21,282	53,485	27,502	21,285	17,968	40,194
6	36,191	32,491	26,551	54,633	21,234	47,565	26,984	21,347	18,022	39,521
7	34,130	32,927	26,548	55,188	21,070	51,224	26,840	21,592	17,966	39,048
8	38,004	32,814	26,555	54,189	21,330	47,833	26,970	21,470	18,047	39,659
9	36,272	32,144	26,546	55,019	21,294	52,592	26,852	23,097	17,995	38,766
10	37,461	33,647	26,556	54,062	21,418	44,765	26,617	21,769	17,979	40,038
11	37,369	32,374	26,557	54,721	21,354	49,345	27,599	21,169	17,988	41,609
12	37,761	32,721	26,561	54,055	20,997	49,873	27,377	21,268	17,984	39,787
13	39,111	32,319	26,557	54,464	21,377	55,045	27,169	21,422	18,074	40,988

Lượt chạy	Khởi tạo phỏng đoán/Chi phí									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
14	35,983	33,696	26,558	55,700	21,560	48,700	27,162	21,989	18,011	41,101
15	37,220	32,624	26,554	53,826	21,503	48,563	27,325	22,272	18,031	38,813
16	37,475	32,309	26,547	53,894	21,187	47,581	26,971	21,528	18,020	39,518
17	36,075	32,276	26,548	54,866	21,564	51,894	27,307	21,043	18,038	39,620
18	42,024	32,865	26,553	55,008	20,927	48,270	26,731	22,351	18,067	41,722
19	44,980	32,997	26,561	54,589	21,026	50,471	27,414	21,806	17,971	38,364
20	38,880	33,162	26,555	55,604	21,331	54,658	27,015	21,808	18,077	39,978
21	37,436	32,405	26,565	54,174	21,096	49,766	26,935	22,832	17,972	39,069
22	40,887	32,070	26,550	53,832	21,429	44,963	27,337	22,314	18,048	40,738
23	34,369	33,871	26,550	54,924	21,538	48,080	27,540	21,097	17,987	40,148
24	36,159	32,281	26,574	54,644	21,256	48,579	26,763	22,600	17,994	40,798
25	33,939	32,360	26,570	53,316	21,728	50,945	26,837	21,666	17,960	39,841
26	38,861	33,202	26,558	54,897	21,217	49,639	27,289	21,587	17,979	40,861
27	38,870	33,113	26,549	54,812	21,394	48,345	26,821	24,154	17,969	39,572
28	37,660	32,376	26,552	54,830	21,422	51,976	27,464	21,047	18,001	39,452
29	44,609	33,487	26,557	54,483	21,023	52,191	27,356	21,675	18,074	39,770
30	34,846	32,529	26,553	54,600	21,357	45,997	27,042	21,385	18,048	39,011
31	38,658	35,028	26,555	54,227	21,531	54,946	27,144	22,624	18,027	38,667
32	40,657	32,255	26,545	54,710	21,696	48,766	27,591	21,207	17,999	38,185
33	36,238	32,237	26,547	55,540	21,481	53,393	27,466	21,233	17,967	39,203
34	34,330	32,414	26,553	53,906	21,338	46,131	27,023	23,508	18,090	41,017
35	38,862	32,476	26,549	53,648	21,340	45,222	27,032	23,215	18,029	39,163
36	37,305	33,311	26,558	53,637	21,389	47,553	27,202	21,606	17,954	40,649
37	38,312	32,397	26,550	53,877	21,546	51,467	26,842	21,959	17,977	40,609
38	38,679	32,879	26,554	53,662	21,285	48,139	27,108	22,570	18,012	39,166
39	37,633	32,354	26,547	54,215	21,491	48,833	26,902	21,928	18,007	39,680
40	37,297	33,641	26,570	54,276	21,160	48,684	27,044	22,136	18,031	39,419
41	43,991	32,174	26,549	54,233	21,506	50,573	27,181	21,749	18,019	39,091
42	34,264	32,757	26,559	55,241	21,169	50,361	27,056	21,589	17,984	39,323
43	41,617	32,210	26,553	53,800	21,212	54,442	26,958	22,309	17,969	38,823
44	36,574	32,560	26,548	53,987	21,597	47,785	27,450	21,516	17,990	40,735
45	35,786	33,083	26,546	54,274	21,368	50,446	27,023	21,319	17,972	39,217
46	36,159	37,572	26,560	53,153	21,700	49,118	27,246	23,159	18,035	39,258
47	46,005	33,018	26,552	53,376	21,212	46,704	27,012	21,987	18,003	40,587
48	37,787	32,727	26,552	54,867	20,985	50,435	27,062	22,015	18,049	41,597
49	44,854	32,826	26,553	54,144	21,568	48,932	27,311	21,165	18,031	41,204
50	36,321	32,456	26,569	53,937	21,615	48,509	27,200	21,265	18,097	41,669

Bảng 4.9: Số lượng điểm trung gian sau giai đoạn khởi tạo phỏng đoán

Lượt chạy	Khởi tạo phỏng đoán/WP									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
1	14	8	3	13	4	16	6	6	3	10
2	19	6	3	13	5	9	7	7	3	8
3	14	7	3	13	5	11	5	8	3	8
4	22	7	3	11	5	14	7	6	3	9
5	21	4	3	13	4	19	7	5	3	8
6	12	7	3	13	6	14	6	5	3	9
7	12	7	3	11	4	17	5	6	3	12
8	16	5	3	11	6	12	5	5	3	10
9	16	4	3	15	5	21	5	9	3	8

Lượt chạy	Khởi tạo phỏng đoán/WP									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
10	15	6	3	10	5	9	5	5	3	11
11	18	6	3	11	5	17	5	5	3	14
12	16	5	3	12	4	12	6	6	3	11
13	15	5	3	11	5	21	5	5	3	11
14	14	7	3	13	4	15	5	6	3	11
15	14	6	3	12	5	9	5	5	3	10
16	18	5	3	13	5	13	5	5	3	8
17	14	5	3	13	4	20	8	6	3	9
18	25	5	3	12	5	15	5	6	3	13
19	32	6	3	12	4	17	6	6	3	10
20	17	6	3	13	6	17	5	5	3	9
21	17	7	3	12	5	14	6	10	3	11
22	19	4	3	15	4	8	6	6	3	11
23	11	6	3	14	6	9	5	7	3	10
24	12	5	3	11	4	14	4	6	3	9
25	13	5	3	11	4	17	7	5	3	10
26	22	8	3	13	4	14	6	5	3	13
27	18	5	3	14	7	12	5	7	3	11
28	18	5	3	11	5	21	5	6	3	9
29	29	8	3	14	4	19	6	5	3	12
30	13	6	3	12	6	11	5	6	3	9
31	18	9	3	12	5	21	5	7	3	9
32	20	6	3	13	4	13	7	5	3	8
33	11	5	3	16	4	18	5	5	3	9
34	14	6	3	12	6	10	4	7	3	11
35	18	5	3	10	4	8	5	10	3	7
36	14	6	3	13	5	10	5	6	3	11
37	19	5	3	10	6	17	5	6	3	12
38	16	5	3	13	4	13	5	8	3	6
39	17	5	3	11	7	13	7	6	3	10
40	18	8	3	14	4	14	5	8	3	10
41	26	6	3	11	4	18	7	6	3	9
42	10	6	3	14	5	14	6	5	3	10
43	24	5	3	13	4	21	7	5	3	9
44	14	5	3	11	4	13	5	5	3	13
45	12	7	3	11	5	16	5	5	3	11
46	12	11	3	12	4	13	5	7	3	10
47	29	6	3	14	4	11	6	7	3	12
48	16	5	3	12	5	13	6	6	3	14
49	26	7	3	12	4	14	8	6	3	13
50	14	7	3	12	6	14	6	6	3	13

Bảng 4.10: Giá trị chi phí sau giai đoạn tìm kiếm của ElasticABC

Lượt chạy	ElasticABC/Chi phí									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
1	31,905	31,566	26,536	50,873	20,531	43,748	25,789	19,909	17,895	36,290
2	30,292	31,659	26,536	50,520	20,532	43,622	25,815	20,065	17,895	36,328
3	30,681	31,538	26,536	50,614	20,531	44,256	25,798	19,899	17,895	36,321
4	30,199	31,549	26,542	51,015	20,533	44,882	25,806	19,937	17,895	36,313
5	30,405	31,629	26,536	50,645	20,531	43,811	25,789	20,012	17,895	36,330

Lượt chạy	ElasticABC/Chi phí									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
6	30,128	31,541	26,536	50,634	20,532	47,082	25,789	19,913	17,895	36,319
7	31,097	31,532	26,536	50,753	20,531	43,543	25,789	19,955	17,933	36,332
8	30,007	31,527	26,536	50,605	20,535	43,282	25,789	19,956	17,895	36,315
9	31,840	31,530	26,536	50,523	20,531	43,842	25,789	19,912	17,896	36,259
10	30,250	31,536	26,536	50,603	20,531	44,219	25,789	20,027	17,895	36,508
11	31,846	31,535	26,536	50,563	20,531	43,246	25,789	19,984	17,895	36,314
12	30,386	31,551	26,536	51,080	20,531	44,210	25,794	20,154	17,895	36,397
13	30,276	31,544	26,536	50,700	20,531	43,610	25,789	19,954	17,896	36,190
14	31,804	31,534	26,536	50,640	20,531	44,199	25,789	19,971	17,895	36,219
15	30,809	31,546	26,536	50,545	20,531	43,434	25,790	20,054	17,895	36,285
16	30,419	31,581	26,536	50,914	20,531	43,605	25,789	19,961	17,895	36,301
17	30,613	31,535	26,536	50,624	20,531	43,427	25,810	19,916	17,895	36,304
18	30,566	31,540	26,536	50,680	20,531	44,571	25,789	19,944	17,896	36,285
19	30,660	31,623	26,542	50,877	20,531	44,825	25,789	19,912	17,895	36,320
20	30,206	31,649	26,536	50,925	20,531	43,859	25,789	19,937	17,895	36,179
21	30,765	31,529	26,536	50,885	20,532	43,721	25,789	19,901	17,895	36,293
22	30,488	31,551	26,537	50,566	20,531	43,762	25,789	19,965	17,949	36,353
23	30,172	31,573	26,536	50,610	20,531	45,116	25,789	19,957	17,895	36,282
24	30,268	31,535	26,536	50,649	20,531	43,911	25,789	20,052	17,895	36,267
25	30,266	31,542	26,536	51,198	20,531	43,614	25,797	20,088	17,896	36,288
26	34,799	31,509	26,536	50,576	20,531	43,604	25,789	19,986	17,895	36,362
27	30,577	31,540	26,536	50,666	20,531	43,671	25,789	19,927	17,895	36,333
28	30,921	31,539	26,536	50,670	20,531	43,779	25,789	19,904	17,895	36,328
29	32,979	31,578	26,536	50,865	20,531	43,977	25,789	19,946	17,895	36,310
30	30,250	31,539	26,536	51,075	20,531	43,633	25,789	19,971	17,895	36,176
31	30,256	31,614	26,537	51,008	20,532	44,429	25,790	20,055	17,895	36,329
32	30,673	31,536	26,536	51,790	20,531	44,140	25,789	19,949	17,895	36,302
33	31,806	31,538	26,536	50,523	20,531	43,697	25,789	19,966	17,895	36,230
34	30,833	31,547	26,536	51,078	20,531	45,538	25,789	19,945	17,895	36,340
35	30,145	31,554	26,536	51,563	20,531	43,716	25,789	19,957	17,895	36,313
36	30,270	31,556	26,536	50,634	20,531	43,455	25,789	19,966	17,895	36,351
37	30,835	31,547	26,536	50,949	20,532	45,037	25,789	19,981	17,895	36,221
38	30,534	31,652	26,536	51,636	20,532	44,574	25,789	19,938	17,895	36,352
39	31,051	31,542	26,536	50,633	20,535	43,800	25,789	20,098	17,895	36,321
40	32,753	31,537	26,536	52,035	20,531	43,534	25,789	19,912	17,895	36,334
41	30,278	31,555	26,536	50,867	20,538	43,274	25,789	19,916	17,895	36,443
42	30,304	31,534	26,536	50,778	20,532	44,124	25,789	19,905	17,895	36,348
43	30,587	31,544	26,536	50,570	20,531	44,592	25,789	20,161	17,897	36,337
44	31,712	31,541	26,536	50,507	20,532	43,744	25,789	20,120	17,895	36,382
45	30,583	31,541	26,536	51,070	20,531	42,966	25,789	20,062	17,895	36,313
46	30,484	31,544	26,538	50,287	20,531	44,067	25,789	19,972	17,895	36,301
47	30,012	31,537	26,536	50,405	20,532	43,370	25,789	19,996	17,898	36,315
48	31,013	31,534	26,536	50,683	20,540	44,050	25,789	19,937	17,895	36,331
49	30,529	31,540	26,536	50,562	20,531	44,387	25,789	19,939	17,895	36,514
50	30,670	31,665	26,536	51,342	20,531	42,799	25,789	19,954	17,895	36,321

Bảng 4.11: Số lượng điểm trung gian sau giai đoạn tìm kiếm của ElasticABC

Lượt chạy	ElasticABC/WP									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
1	11	3	3	5	3	5	4	5	3	4

Lượt chạy	ElasticABC/WP									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
2	5	4	3	5	3	8	4	5	3	4
3	6	3	3	5	3	5	4	5	3	4
4	5	3	3	5	3	8	4	4	3	4
5	5	4	3	6	3	7	4	5	3	4
6	6	3	3	5	3	13	4	5	3	4
7	9	5	3	5	3	6	4	6	3	4
8	5	5	3	5	3	5	4	4	3	4
9	6	4	3	5	3	7	4	5	3	4
10	5	5	3	6	3	6	4	4	3	4
11	10	5	3	6	3	5	4	5	3	4
12	5	3	3	7	3	7	4	4	3	4
13	5	3	3	5	3	7	4	5	3	4
14	10	4	3	5	3	6	4	4	3	4
15	4	3	3	5	3	5	4	4	3	4
16	5	4	3	5	3	5	4	4	3	4
17	5	3	3	5	3	6	4	5	3	4
18	5	3	3	5	3	6	4	5	3	4
19	4	4	3	7	3	10	4	5	3	4
20	5	4	3	5	3	6	4	4	3	4
21	4	4	3	6	3	5	4	5	3	4
22	5	3	3	5	3	7	4	4	3	4
23	5	4	3	5	3	9	4	4	3	4
24	5	3	3	5	3	5	4	5	3	4
25	5	3	3	6	3	5	4	4	3	4
26	10	5	3	5	3	5	4	5	3	4
27	4	3	3	6	3	6	4	5	3	4
28	7	5	3	5	3	5	4	5	3	4
29	13	4	3	5	3	7	4	4	3	4
30	5	3	3	5	3	5	4	4	3	4
31	5	4	3	6	3	7	4	4	3	4
32	4	3	3	7	3	9	4	5	3	4
33	6	5	3	6	3	7	4	4	3	4
34	5	4	3	5	3	8	4	5	3	4
35	5	4	3	6	3	5	4	4	3	4
36	5	3	3	5	3	6	4	4	3	5
37	5	4	3	5	3	11	4	4	3	4
38	4	4	3	7	3	7	4	5	3	4
39	9	3	3	5	3	6	4	4	3	4
40	12	4	3	8	3	6	4	5	3	4
41	5	4	3	6	3	5	4	5	3	4
42	5	3	3	5	3	5	4	5	3	4
43	6	3	3	5	3	9	4	4	3	4
44	8	3	3	5	3	6	4	4	3	4
45	5	3	3	5	3	6	4	4	3	6
46	4	3	3	5	3	6	4	5	3	4
47	6	4	3	5	3	6	4	5	3	5
48	6	3	3	5	3	5	4	4	3	4
49	5	3	3	5	3	6	4	5	3	4
50	4	4	3	6	3	5	4	5	3	4

Các chỉ số thống kê cho bộ số liệu 50 lượt chạy tương ứng cho hai trường hợp TH #1 và TH #2 bao gồm: thứ nhất là giá trị trung bình μ , độ lệch chuẩn σ , hệ số biến

thiên (CV %), trung vị, tứ phân vị 1 (Q1) với 3 (Q3), khoảng tứ phân vị (IQR), khoảng tin cậy 95% (2σ) cho giá trị chi phí ở thời điểm hoàn thành khởi tạo phỏng đoán và thời điểm giai đoạn tìm kiếm kết thúc cho từng phần tử robot; thứ hai là giá trị trung bình μ , độ lệch chuẩn σ , hệ số biến thiên (CV %), trung vị, yếu vị (mode), tỷ lệ hội tụ (%), khoảng bách phân vị (P5 P95) cho số lượng điểm trung gian cho thời điểm hoàn thành khởi tạo phỏng đoán và thời điểm giai đoạn tìm kiếm kết thúc cho từng phần tử robot. Bản chất liên tục của giá trị chi phí khác biệt với gián đoạn của số lượng điểm trung gian tạo nên sự khác biệt trong chỉ số thống kê đã chọn ở trên. Để so sánh kết quả giữa hai thời điểm, tỷ lệ giảm chi phí trung bình (%) được tính cho giá trị chi phí và tỷ lệ WP giảm theo trung vị (%) được tính cho số lượng điểm trung gian. Bảng 4.12 cho trường hợp TH #1, bộ dữ liệu #A3, và Bảng 4.13 cho trường hợp TH #2, bộ dữ liệu #A4. Xu hướng cải thiện tương tự như Bộ dữ liệu #A1 và Bộ dữ liệu #A2 đã được thể hiện rõ qua số liệu thống kê; cho thấy hai thử nghiệm đơn lẻ không phải là trường hợp ngoại lai mà có sự nhất quán cao theo quy luật hoạt động của thuật toán ElasticABC.

Bảng 4.12: Chỉ số thống kê chi phí và số lượng điểm trung gian, Bộ dữ liệu #A3

Chỉ số thống kê	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
<i>Chi phí sau khi khởi tạo phỏng đoán</i>										
Trung bình (μ)	38,238	32,898	26,556	54,383	21,340	49,502	27,130	21,933	18,010	39,874
Độ lệch chuẩn (σ)	2,966	0,896	0,009	0,589	0,196	2,566	0,236	0,694	0,036	0,903
Hệ số biến thiên (CV %)	7,757	2,723	0,034	1,083	0,917	5,184	0,871	3,165	0,201	2,266
Trung vị	37,554	32,691	26,553	54,275	21,339	48,883	27,071	21,759	18,005	39,670
Tứ phân vị 1 (Q1)	36,167	32,364	26,549	53,914	21,212	47,895	26,970	21,394	17,980	39,175
Tứ phân vị 3 (Q3)	39,053	33,105	26,558	54,849	21,500	50,908	27,322	22,342	18,031	40,639
Khoảng tứ phân vị (IQR)	2,886	0,742	0,009	0,936	0,288	3,014	0,351	0,947	0,051	1,464
Khoảng tin cậy 95% (2σ)	(32,305 44,170)	(31,107 34,690)	(26,538 26,574)	(53,205 55,560)	(20,948 21,731)	(44,370 54,635)	(26,657 27,603)	(20,545 23,322)	(17,938 18,083)	(38,067 41,681)
<i>Chi phí sau khi kết thúc tìm kiếm</i>										

Chỉ số thống kê	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Trung bình (μ)	30,824	31,557	26,536	50,820	20,532	43,987	25,791	19,976	17,897	36,316
Độ lệch chuẩn (σ)	0,873	0,037	0,001	0,351	0,002	0,703	0,005	0,066	0,009	0,064
Hệ số biến thiên (CV %)	2,834	0,117	0,005	0,691	0,008	1,598	0,021	0,331	0,051	0,176
Trung vị	30,572	31,541	26,536	50,675	20,531	43,790	25,789	19,956	17,895	36,317
Tứ phân vị 1 (Q1)	30,271	31,536	26,536	50,603	20,531	43,606	25,789	19,937	17,895	36,291
Tứ phân vị 3 (Q3)	30,899	31,555	26,536	50,943	20,532	44,217	25,789	19,993	17,895	36,334
Khoảng tứ phân vị (IQR)	0,628	0,018	0,000	0,340	0,001	0,611	0,000	0,056	0,000	0,043
Khoảng tin cậy 95% (2 σ)	(29,077 32,571)	(31,483 31,631)	(26,534 26,539)	(50,118 51,523)	(20,528 20,535)	(42,581 45,393)	(25,780 25,801)	(19,844 20,108)	(17,879 17,915)	(36,188 36,444)
Tỷ lệ giảm chi phí trung bình (%)	19,39	4,08	0,07	6,55	3,79	11,14	4,94	8,92	0,63	8,92
<i>Số lượng WP sau khi khởi tạo phỏng đoán</i>										
Trung bình (μ)	17,280	6,020	3,000	12,360	4,780	14,420	5,640	6,120	3,000	10,220
Độ lệch chuẩn (σ)	5,000	1,334	0,000	1,338	0,855	3,656	0,933	1,243	0,000	1,792
Hệ số biến thiên (CV %)	28,936	22,160	0,000	10,826	17,894	25,351	16,542	20,314	0,000	17,535
Trung vị	16,000	6,000	3,000	12,000	5,000	14,000	5,000	6,000	3,000	10,000
Yếu vị	14	5	3	13	4	14	5	6	3	9
Tỷ lệ hội tụ (%)	18,0	36,0	100,0	28,0	46,0	18,0	52,0	38,0	100,0	22,0
Khoảng bách phân vị (P5 P95)	[11 27]	[4 8]	[3 3]	[10 14]	[4 6]	[9 21]	[5 7]	[5 8]	[3 3]	[8 13]
<i>Số lượng WP sau khi kết thúc tìm kiếm</i>										
Trung bình (μ)	5,960	3,660	3,000	5,440	3,000	6,460	4,000	4,560	3,000	4,080
Độ lệch chuẩn (σ)	2,181	0,710	0,000	0,726	0,000	1,700	0,000	0,535	0,000	0,337
Hệ số biến thiên (CV %)	36,600	19,405	0,000	13,337	0,000	26,309	0,000	11,736	0,000	8,261
Trung vị	5,000	4,000	3,000	5,000	3,000	6,000	4,000	5,000	3,000	4,000
Yếu vị	5	3	3	5	3	5	4	5	3	4
Tỷ lệ hội tụ (%)	50,0	48,0	100,0	68,0	100,0	34,0	100,0	52,0	100,0	94,0

Chỉ số thống kê	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Khoảng bách phân vị (P5 P95)	[4 10]	[3 5]	[3 3]	[5 7]	[3 3]	[5 9]	[4 4]	[4 5]	[3 3]	[4 4]
Tỷ lệ WP giảm theo trung vị (%)	68,8	33,3	0,0	58,3	40,0	57,1	20,0	16,7	0,0	60,0

Bảng 4.13: Chỉ số thống kê chi phí và số lượng điểm trung gian, Bộ dữ liệu #A4

Chỉ số thống kê	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
<i>Chi phí sau khi khởi tạo phỏng đoán</i>										
Trung bình (μ)	37,887	26,435	29,116	41,966	29,622	22,283	28,296	55,983	43,245	36,708
Độ lệch chuẩn (σ)	5,287	0,165	1,718	1,972	2,119	1,185	0,481	0,292	0,651	0,061
Hệ số biến thiên (CV %)	13,954	0,624	5,899	4,699	7,154	5,319	1,700	0,521	1,506	0,166
Trung vị	37,511	26,368	28,927	41,364	29,426	22,096	28,304	56,016	43,058	36,687
Tứ phân vị 1 (Q1)	34,648	26,353	27,802	40,599	28,133	21,426	28,005	55,731	42,751	36,660
Tứ phân vị 3 (Q3)	40,913	26,383	30,275	42,789	30,844	22,861	28,637	56,215	43,570	36,738
Khoảng tứ phân vị (IQR)	6,265	0,030	2,473	2,190	2,711	1,435	0,632	0,484	0,819	0,078
Khoảng tin cậy 95% (2σ)	(27,313 48,461)	(26,105 26,765)	(25,681 32,551)	(38,022 45,910)	(25,383 33,860)	(19,912 24,653)	(27,334 29,258)	(55,400 56,566)	(41,942 44,547)	(36,585 36,830)
<i>Chi phí sau khi kết thúc tìm kiếm</i>										
Trung bình (μ)	30,011	26,314	24,214	38,330	25,180	19,620	26,269	54,216	42,214	36,374
Độ lệch chuẩn (σ)	1,693	0,004	0,502	0,332	0,627	0,243	0,094	0,060	0,030	0,003
Hệ số biến thiên (CV %)	5,641	0,017	2,075	0,866	2,489	1,241	0,358	0,110	0,071	0,009
Trung vị	29,840	26,313	24,127	38,252	24,911	19,562	26,227	54,208	42,202	36,373
Tứ phân vị 1 (Q1)	28,807	26,313	23,856	38,198	24,730	19,417	26,184	54,172	42,191	36,373
Tứ phân vị 3 (Q3)	30,674	26,313	24,538	38,293	25,432	19,828	26,363	54,245	42,229	36,374
Khoảng tứ phân vị (IQR)	1,867	0,000	0,682	0,095	0,702	0,411	0,179	0,073	0,038	0,001

Chỉ số thống kê	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
Khoảng tin cậy 95% (2 σ)	(26,625 33,397)	(26,305 26,323)	(23,209 25,218)	(37,666 38,994)	(23,926 26,434)	(19,133 20,107)	(26,081 26,457)	(54,097 54,335)	(42,154 42,274)	(36,367 36,381)
Tỷ lệ giảm chi phí trung bình (%)	20,79	0,46	16,84	8,66	14,99	11,95	7,16	3,16	2,38	0,91
<i>Số lượng WP sau khi khởi tạo phỏng đoán</i>										
Trung bình (μ)	23,400	3,340	14,080	11,480	12,520	9,580	9,140	10,580	6,280	3,380
Độ lệch chuẩn (σ)	7,697	0,620	2,652	3,208	3,413	2,192	1,456	1,372	1,150	0,485
Hệ số biến thiên (CV %)	32,892	18,563	18,836	27,942	27,262	22,878	15,932	12,972	18,306	14,361
Trung vị	23,000	3,000	14,000	10,000	12,000	9,000	9,000	11,000	6,000	3,000
Yếu vị	25	3	12	10	12	8	9	11	6	3
Tỷ lệ hội tụ (%)	10,0	74,0	26,0	22,0	18,0	26,0	34,0	34,0	40,0	62,0
Khoảng bách phân vị (P5 P95)	[13 35]	[3 5]	[11 19]	[8 18]	[7 17]	[7 14]	[7 11]	[8 13]	[5 8]	[3 4]
<i>Số lượng WP sau khi kết thúc tìm kiếm</i>										
Trung bình (μ)	14,520	3,040	7,100	4,480	6,120	5,840	4,080	5,100	4,100	3,000
Độ lệch chuẩn (σ)	3,833	0,196	2,427	1,330	1,925	1,419	0,337	0,300	0,300	0,000
Hệ số biến thiên (CV %)	26,396	6,446	34,182	29,693	31,454	24,303	8,261	5,882	7,317	0,000
Trung vị	14,000	3,000	7,000	4,000	6,000	6,000	4,000	5,000	4,000	3,000
Yếu vị	13	3	4	4	5	5	4	5	4	3
Tỷ lệ hội tụ (%)	20,0	96,0	20,0	78,0	26,0	28,0	94,0	90,0	90,0	100,0
Khoảng bách phân vị (P5 P95)	[9 21]	[3 3]	[4 10]	[4 6]	[3 9]	[4 8]	[4 4]	[5 6]	[4 5]	[3 3]
Tỷ lệ WP giảm theo trung vị (%)	39,13	0,00	50,00	60,00	50,00	33,33	55,56	54,55	33,33	0,00

Để xem xét về khả năng tối ưu chi phí đường đi, xu hướng giảm chung ở tất cả phần tử robot cũng được minh chứng rất rõ ràng qua chỉ số thống kê: chi phí trung bình, trung vị ở thời điểm sau khi kết thúc tìm kiếm đều giảm so với thời điểm sau khi khởi tạo phỏng đoán. Khi xem xét cho từng phần tử, mức độ cải thiện đã có sự khác biệt rõ ràng, thể hiện ở chỉ số tỷ lệ giảm chi phí trung bình. Một số đường đi

phức tạp có chi phí khởi tạo lớn đã được tối ưu mạnh mẽ ở giai đoạn tìm kiếm. Chi phí giảm tới 19,39% với R1, 11,14% với R2 trong bộ dữ liệu #A3; giảm tới 20,79% với R1, 16,84% với R3, 14,99% với R5 trong bộ dữ liệu #A4. Ngược lại, chi phí giảm rất nhỏ 0,07% với R3, 0,63% với R9 trong bộ dữ liệu #A3; giảm 0,46% với R2, 0,91% với R10 trong bộ dữ liệu #A4. Chúng là những phần tử robot có đường đi được khởi tạo tương đối tốt, kết quả tìm kiếm của thuật toán có mức giảm chi phí không đáng kể. Thuật toán có khả năng "nhận thức" và duy trì được mức chi phí tiệm cận tối ưu mà không làm thay đổi cấu trúc tốt đã có.

Hệ số biến thiên (CV) của chi phí sau khi khởi tạo phỏng đoán, tính theo chỉ số độ lệch chuẩn (σ), dao động trong khoảng rộng: 8/10 phần tử có giá trị từ gần 1% đến gần 8% (7,757% của R1 và 5,184 của R6), 2/10 phần tử có giá trị không quá 0,2% R2 trong bộ dữ liệu #A3; 7/10 phần tử có giá trị từ gần 1% đến gần 14% (13,954% của R1, 5,899% của R3 và 7,154% của R5), 3/10 phần tử có giá trị không quá 0,7%. Hệ số biến thiên (CV) của chi phí sau khi kết thúc tìm kiếm nhỏ hơn rất nhiều, dao động trong khoảng hẹp: 8/10 phần tử có giá trị dưới 0,7%, 2/10 phần tử có giá trị lớn hơn 1% (2,834% của R1 và 1,598% của R6) trong bộ dữ liệu #A3; 5/10 phần tử có giá trị dưới 0,4%, 5/10 phần tử có giá trị trong khoảng 0,8% đến 5,7% (5,641% của R1, 2,075% của R3 và 2,489% của R5) trong bộ dữ liệu #A4. Bên cạnh đó, khoảng tin cậy 95% (2σ) của chi phí sau khi kết thúc tìm kiếm đã thu hẹp rõ ràng so với chi phí sau khi khởi tạo phỏng đoán.

Như vậy, thuật toán ElasticABC đã khám phá và tìm ra các đường đi ngắn (tốn ít chi phí hơn) rất nhiều so với những kết quả phỏng đoán ban đầu; thể hiện khả năng vượt trội trong việc tinh chỉnh và tối ưu chi phí di chuyển; không chỉ khắc phục được các đường đi chưa đạt chất lượng mà còn đảm bảo sự hội tụ chính xác đối với các đường đi đơn giản. Tính ổn định của thuật toán cũng được thể hiện qua xu hướng hội tụ của bộ dữ liệu #A3 và #A4, khi kết thúc vòng lặp của thuật toán, quần thể ong đều đồng thuận tìm về một vùng kết quả tối ưu rất sát nhau.

Để xem xét độ phức tạp của đường đi và hoạt động điều chỉnh số lượng điểm trung gian, xu hướng giảm của chỉ số giá trị trung bình của số lượng điểm trung gian

ở thời điểm sau khi kết thúc giai đoạn tìm kiếm so với thời điểm sau khi khởi tạo đã minh chứng rõ ràng ở các phần tử robot. Những phần tử có chỉ số giá trị trung bình rất cao ở giai đoạn khởi tạo (17,28 cho R1, 14,42 cho R6 của bộ dữ liệu #A3; 23,4 cho R1, 14,08 cho R3 của bộ dữ liệu #A4) sau khi hoàn thành giai đoạn tìm kiếm, số lượng điểm trung gian đã giảm xuống (5,96 cho R1, 6,46 cho R6 của bộ dữ liệu #A3; 14,52 cho R1, 7,10 cho R3 của bộ dữ liệu #A4). Giá trị trung vị cũng giảm tương tự ở tất cả các phần tử robot đã hỗ trợ cho xu hướng nói trên. Bên cạnh đó, tỷ lệ giảm (so sánh giữa giá trị trung vị) cũng có sự phân hoá rõ ràng, phản ánh mức độ triệt tiêu lượng điểm trung gian. Cụ thể, 8/10 phần tử có mức độ giảm trong khoảng 15% cho đến gần 70%, 2/10 phần tử không giảm (R3 và R9 giữ tỷ lệ 0%) trong bộ dữ liệu #A3; 8/10 phần tử có mức độ giảm trong khoảng 33% cho đến 60%, 2/10 phần tử không giảm (R2 và R10 giữ tỷ lệ 0%) trong bộ dữ liệu #A4.

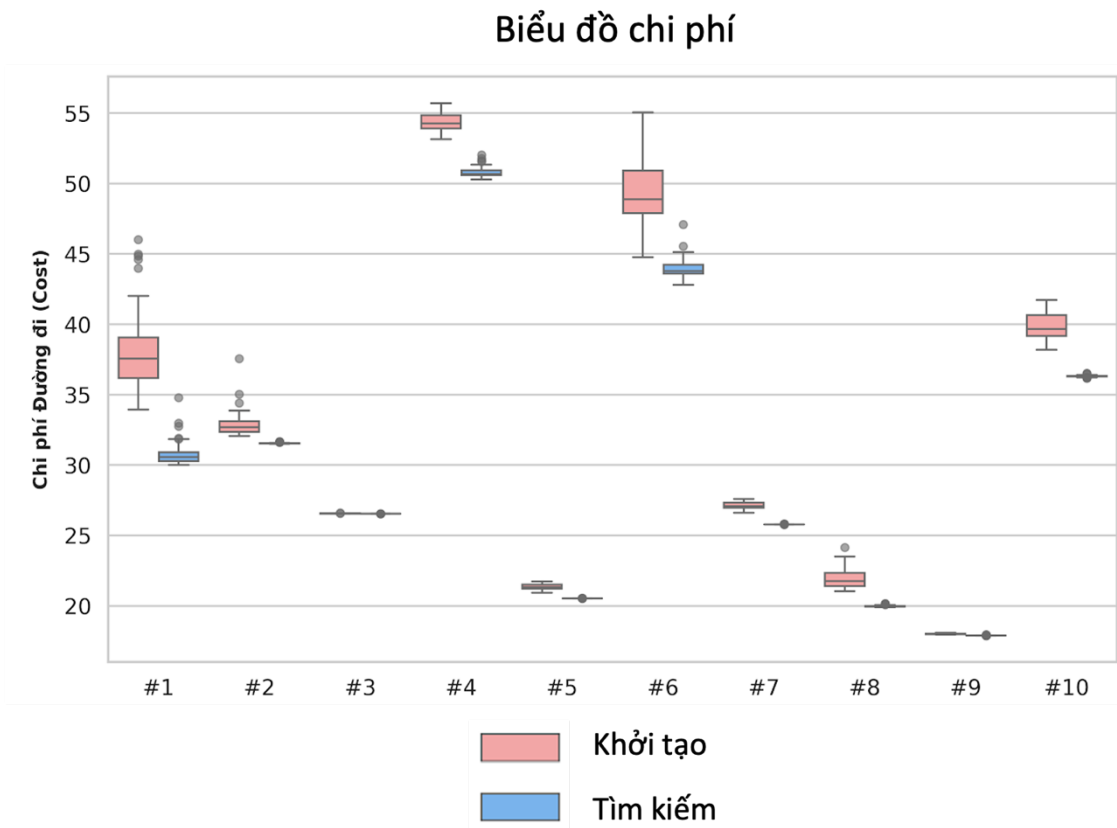
Phối hợp giữa chỉ số yếu vị (mode), tỷ lệ hội tụ và khoảng bách phân vị (P5 P95) cho số lượng điểm trung gian ở thời điểm sau khi kết thúc giai đoạn tìm kiếm đã minh chứng cho cấu trúc đường đi tối ưu ở một số phần tử robot: 5/10 phần tử (R3, R5, R7, R9 và R10) đạt tỷ lệ hội tụ trên 94% trong bộ dữ liệu #A3; 3/10 phần tử (R2, R7 và R10) đạt tỷ lệ hội tụ trên 94% trong bộ dữ liệu #A4. Các kết quả gần tối ưu đạt được ở 5/10 phần tử robot trong bộ dữ liệu #A3 và 5/10 phần tử robot trong bộ dữ liệu #A4.

Như vậy, trong khi giai đoạn khởi tạo phỏng đoán đã thiết lập đường đi ban đầu với cấu trúc dư thừa điểm để đảm bảo an toàn và chậm, hoạt động của cặp toán tử đối ngẫu, Co và Dẫn, ở giai đoạn tìm kiếm ong thợ đã tham gia tích cực vào việc định hình cấu trúc đường đi tối ưu với số lượng điểm trung gian giảm xuống rõ rệt. Đồng thời, lượng giảm cuối cùng có sự khác biệt rõ ràng cho từng đường đi riêng lẻ.

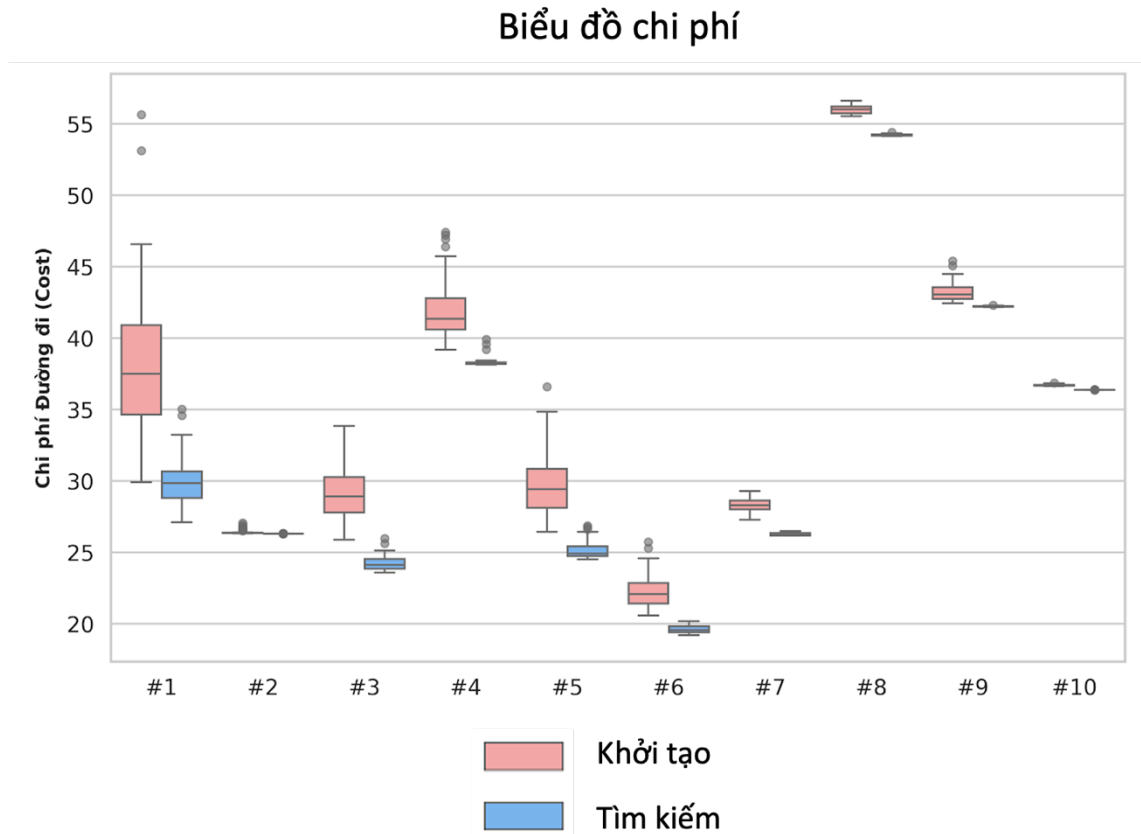
Trong bộ dữ liệu #A4, các phần tử R1 và R3 có tỷ lệ hội tụ 20% cũng với khoảng bách phân vị (P5 P95) lần lượt là [9 21] và [4 10], là những ngoại lệ khi kết quả tìm kiếm có sự phân tán. Kết hợp với độ lệch chuẩn (σ) và hệ số biến thiên CV tương ứng, kết quả đường đi đã phản ánh sự tồn tại của nhiều giải pháp khả thi với chi phí chênh lệch đáng kể. Thuật toán ElasticABC đã khám phá ra những đường đi khác nhau trong mỗi lần chạy và tất cả chúng đều hợp lệ. Mặc dù số lượng điểm trung gian

còn ở mức cao, thể hiện qua giá trị trung bình 14,55 điểm; chi phí trung bình đã giảm đáng kể so với khởi tạo (từ 37,8 xuống 30,0). Điều này khẳng định thuật toán luôn cải thiện được chất lượng giải pháp đầu vào bất chấp độ khó khăn của môi trường, bao gồm loại bỏ điểm trung gian không cần thiết và giảm chi phí.

Để minh chứng cho những nhận định trên, Hình 4.5 và Hình 4.6 lần lượt minh hoạ biểu đồ hộp của chi phí cho TH #1 và TH #2, phản ánh sự cải thiện rõ ràng về chi phí ở tất cả 10 phần tử robot. Trong đó, chi phí ở thời điểm sau khi khởi tạo là hộp màu hồng, độ cao lớn biểu thị sự phân tán của chi phí sau khi khởi tạo; chi phí ở thời điểm sau khi kết thúc giai đoạn tìm kiếm là hộp màu xanh, luôn nằm thấp hơn hộp hồng; độ cao giảm rõ rệt; một số phần tử có độ cao hộp xanh rất mỏng, cho thấy quần thể đã đạt được sự hội tụ gần như tuyệt đối với mức chi phí cực kỳ đồng đều.



Hình 4.5: Biểu đồ hộp của chi phí cho TH #1

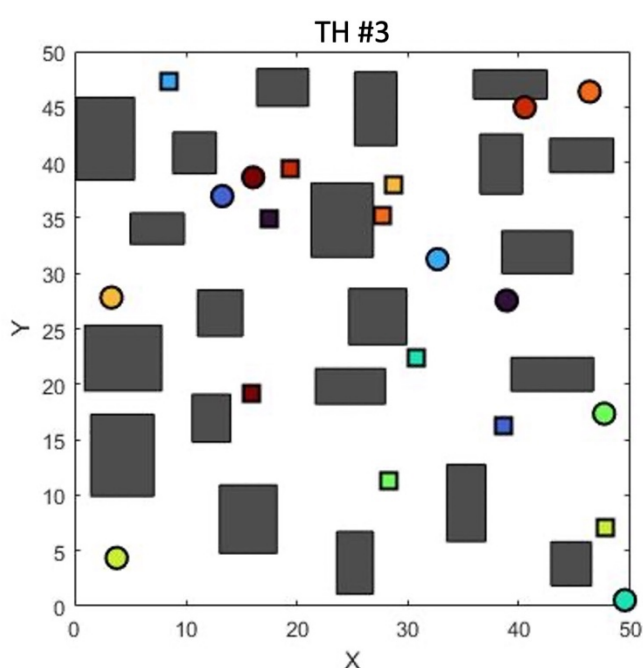


Hình 4.6: Biểu đồ hộp của chi phí cho TH #2

Với kết quả đánh giá về khả năng tối ưu chi phí đường đi cũng như cải thiện cấu trúc đường đi thông qua điều chỉnh số lượng điểm trung gian, dữ liệu của TH #1 và TH #2 cho thấy sự tương quan rất cao. Đầu tiên là khả năng cải thiện các kết quả khởi tạo xấu để cho ra được "nghiệm đầu ra" rất ổn định về chi phí. Tiếp đến sự tương đồng trong tỷ lệ giảm ở cả giá trị trung bình và trung vị, ước tính chi phí tổng thể được đảm bảo cắt giảm từ 15% đến hơn 20%. Thuật toán ElasticABC đã hoạch định hiệu quả đường đi cho các phần tử robot: tối ưu chi phí và rút gọn cấu trúc đường đi cho từng phần tử robot. Thuật toán cũng thể hiện được khả năng hội tụ và tính ổn định, khi đầu vào khởi tạo phỏng đoán có thể phân tán nhưng đầu ra luôn là kết quả hội tụ về một vùng hẹp với độ lệch chuẩn thấp. Việc loại bỏ các điểm trung gian "không cần thiết" cũng góp phần giảm chi phí hàm mục tiêu của từng phần tử.

Xét trường hợp môi trường biến đổi (TH #3) với điều kiện hoàn toàn khác về mật độ vật cản và kích thước vật cản khác so với TH #1 và TH #2. Hình 4.7 minh họa

về cấu hình vị trí vật cản trong TH #3. Các phần tử robot cũng được khởi tạo vị trí bắt đầu và vị trí đích đến hoàn toàn ngẫu nhiên trong vùng không gian trống của bản đồ. Bảng 4.14 tổng hợp giá trị chi phí và số lượng điểm trung gian ở thời điểm hoàn thành khởi tạo phỏng đoán và thời điểm kết thúc giai đoạn tìm kiếm cho từng phần tử robot. Hình 4.8 thể hiện đồ thị đường đi của 10 phần tử, là kết quả hoạch định của thuật toán ElasticABC, hình tròn là điểm bắt đầu, hình vuông là điểm kết thúc, mỗi màu biểu thị hành trình đã tìm được của một phần tử robot

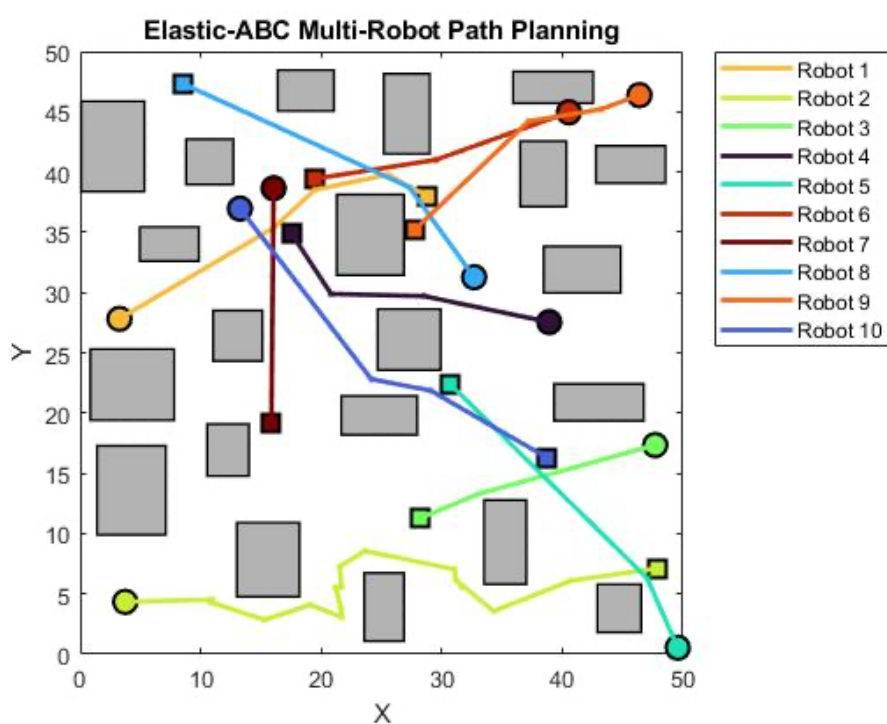


Hình 4.7: Kịch bản thử nghiệm TH #3

Bảng 4.14: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A5

Thứ tự robot	Khởi tạo phỏng đoán		ElasticABC		Mức giảm	
	Chi phí	WP	Chi phí	WP	% chi phí	WP
R1	33,717	12	29,408	6	12,78	6
R2	69,163	46	53,178	22	23,11	24
R3	20,465	3	20,395	3	0,34	0

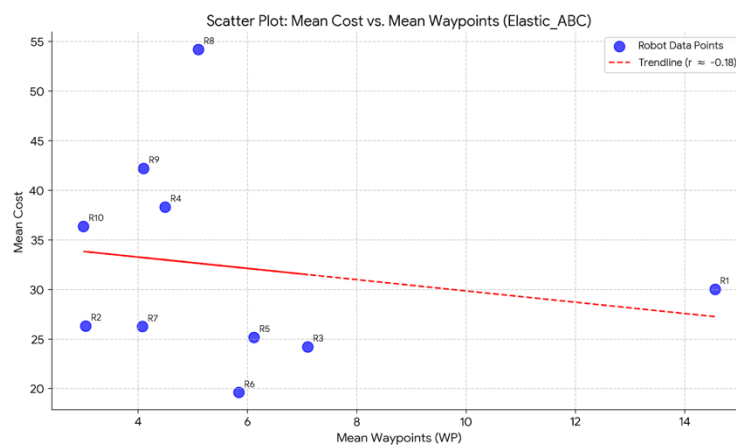
Thứ tự robot	Khởi tạo phỏng đoán		ElasticABC		Mức giảm	
	Chi phí	WP	Chi phí	WP	% chi phí	WP
R4	100,000	51	24,280	4	-	47
R5	29,400	4	29,183	3	0,74	1
R6	21,946	3	21,898	3	0,22	0
R7	19,531	2	19,531	2	0,00	0
R8	30,822	6	29,923	3	2,92	3
R9	26,389	6	22,511	4	14,70	2
R10	100,000	52	33,969	4	-	48
Tổng	451,433		284,276			



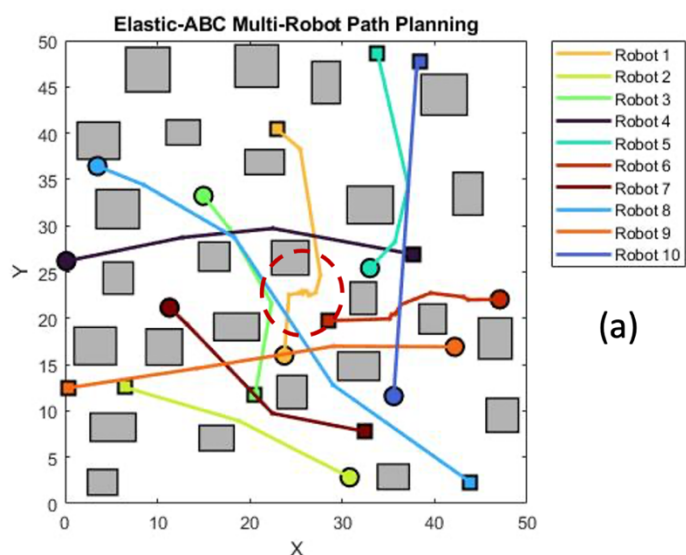
Hình 4.8: Kết quả đường đi tối ưu cho TH #3

Số liệu về chi phí và số lượng điểm trung gian của TH #3 khẳng định sự nhất quán trong cải thiện cấu trúc và chất lượng giải pháp của thuật toán ElasticABC. Xu hướng cắt giảm điểm trung gian đã diễn ra ở tất cả 10 phần tử robot, tỷ lệ giảm trung bình điểm trung gian lên tới hơn 39%. Bên cạnh đó, tỷ lệ cải thiện trung bình của các phần tử robot ở mức khoảng 19,66% trong khi tỷ lệ cải thiện tổng chi phí là 37,03% (từ 451,433 xuống còn 284,276). Thuật toán không chỉ tìm kiếm đường đi ngắn nhất thể hiện qua nỗ lực tối ưu chi phí mà còn tạo ra đường đi thích nghi theo điều kiện môi trường thông qua cơ chế co dãn để tối ưu cấu trúc. Xem xét chi tiết hơn, dữ liệu của phần tử R4 và R10 cho thấy năng lực cải thiện chất lượng giải pháp đầu vào bất chấp độ khó khăn của môi trường của ElasticABC. Từ hiện trạng khởi tạo của hai phần tử là kết quả với mức chi phí và số lượng điểm trung gian rất lớn, giải thuật đã tìm kiếm được giải pháp tốt hơn rất nhiều: R4 có chi phí giảm từ 100 xuống còn 24,28, số lượng điểm trung gian giảm từ 51 xuống 4; tương tự với R10 có 100 xuống 33,969 và 52 xuống 4. Thuật toán tiếp tục cải thiện được chất lượng của giải pháp đường đi ban đầu ngay cả khi gặp phải kịch bản khởi tạo không thành công do môi trường có độ khó cao. Dữ liệu thử nghiệm từ kịch bản TH #3 đã minh chứng tính bền vững của thuật toán, hoạt động hiệu quả trong điều kiện môi trường biến đổi.

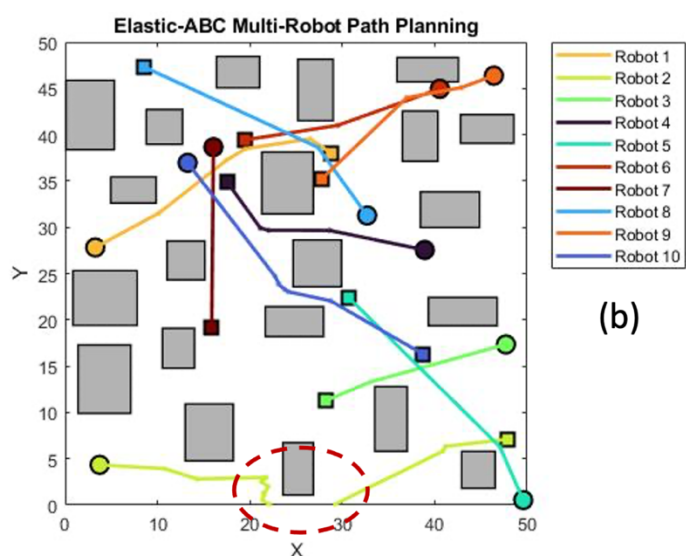
Số liệu của phần tử R1, thể hiện trong Bảng 4.13, là ngoại lệ quan trọng khi xem xét về số lượng điểm trung gian trong giải pháp đường đi tìm được bởi thuật toán ElasticABC. Chúng là các đường đi có độ dài quãng đường không lớn so với phần tử robot khác nhưng lại có số lượng điểm trung gian rất lớn, như biểu đồ phân tán minh họa trong Hình 4.9. Một số kết quả thử nghiệm đơn lẻ cụ thể hơn, thể hiện qua vùng đánh dấu trong Hình 4.10, đã bộc lộ hiện tượng gấp khúc trong đường đi. Cụ thể hơn, phần tử robot R1 (Hình 4.10.a) có 13 điểm trung gian, phần tử robot R2 (Hình 4.10.b) có 19 điểm trung gian; tỷ trọng lớn điểm trung gian lại tập trung trong khoảng không gian rất nhỏ, biểu thị đường đi tại đó đã đổi hướng liên tục.



Hình 4.9: Biểu đồ phân tán của trung bình chi phí và số lượng điểm trung gian



(a)



(b)

Hình 4.10: Hiện tượng gấp khúc trong đường đi

Xem xét hoạt động của kỹ thuật DeZigzagPath trong kịch bản thử nghiệm cụ thể TH #4 và TH #5; thuật toán ElasticABC đã hoàn thành việc tìm kiếm đường đi gần tối ưu và DeZigzagPath được dùng để xử lý hiện tượng gấp khúc. Bảng 4.15 tổng hợp số liệu về chi phí (độ dài đường đi) cho ba giai đoạn: khởi tạo, kết quả của thuật toán ElasticABC, và kết quả sau khi áp dụng kỹ thuật DeZigzagPath, xét cho trường hợp TH #4, Bộ dữ liệu #A6. Tương tự là Bảng 4.16 xét cho trường hợp TH #5, Bộ dữ liệu #A7. Hình 4.11 và Hình 4.12 minh họa đồ thị đường đi của 10 phần tử robot cho hai kịch bản thử nghiệm của trước và sau khi áp dụng DeZigzagPath, hình tròn là điểm bắt đầu, hình vuông là điểm kết thúc, mỗi màu biểu thị hành trình đã tìm được của một phần tử robot.

Đối chiếu hình ảnh đường đi của các robot cho thấy DeZigzagPath đã không thay đổi cấu trúc toàn cục của đường đi trong khi số liệu thu được đã minh chứng rõ ràng về hoạt động loại bỏ điểm trung gian thừa và duỗi thẳng các đoạn gấp khúc không cần thiết. Số liệu nổi bật nhất là của hai quỹ đạo "phức tạp": phần tử R2 trong TH #4 với 25 điểm trung gian đã giảm xuống chỉ còn 7 điểm, phần tử robot R1 trong TH #5 với 19 điểm trung gian đã giảm xuống chỉ còn 4 điểm. Độ dài đường đi đã giảm rõ rệt tương ứng 12% với R2 (từ 56,777 xuống còn 49,904) và 22% với R1 (từ 34,497 xuống còn 26,771). Bên cạnh đó, DeZigzagPath đã tinh chỉnh cho các phần tử R3, R5, và R6 trong TH #5, cắt giảm số lượng đáng kể điểm trung gian, 2 đến 3 điểm, trong khi độ dài đường đi giảm không đáng kể, dưới 1%. Cuối cùng, số liệu định lượng của các phần tử robot còn lại khẳng định tính chất bảo toàn của DeZigzagPath đối với những đường đi đã tối ưu, quan sát định tính qua đồ thị đường đi. Như vậy, dữ liệu thực nghiệm đã cho thấy DeZigzagPath là một bước xử lý hậu kỳ cần thiết cho những đường đi gấp khúc.

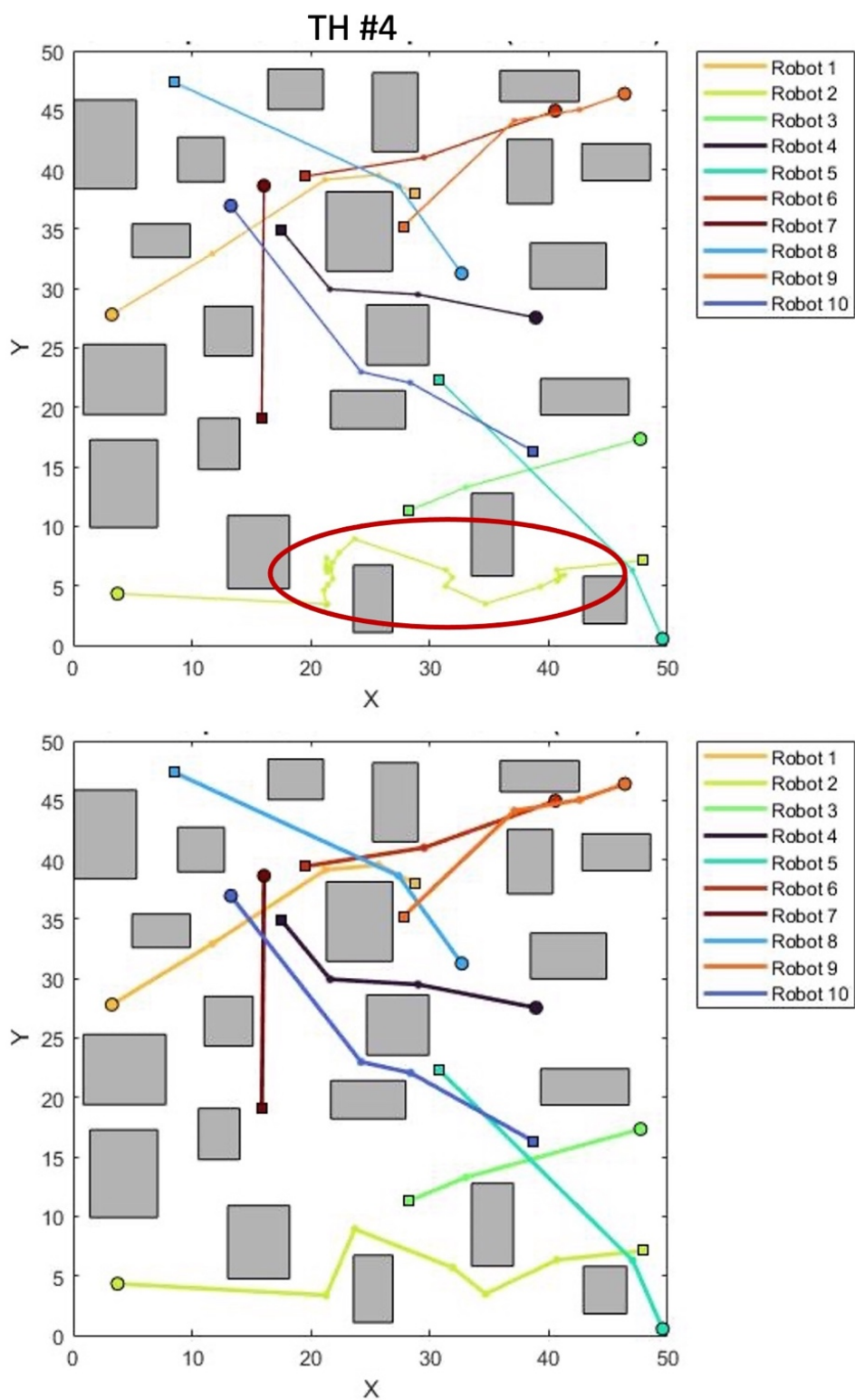
Bảng 4.15: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A6

Thứ tự robot	Khởi tạo phỏng đoán		ElasticABC		DeZigzagPath	
	Chi phí	WP	Chi phí	WP	Chi phí	WP
R1	32,483	12	29,199	5	29,199	5

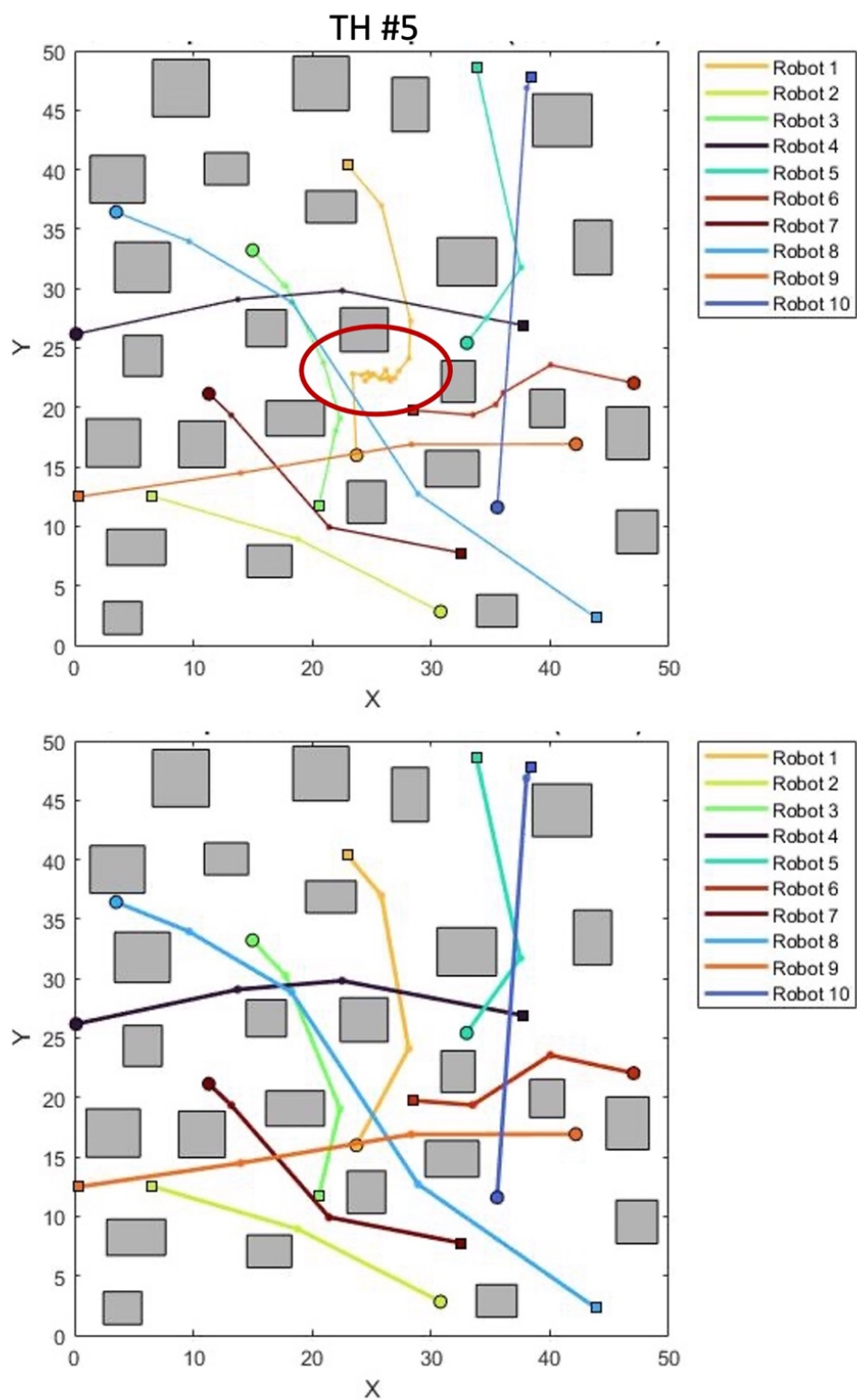
Thứ tự robot	Khởi tạo phỏng đoán		ElasticABC		DeZigzagPath	
	Chi phí	WP	Chi phí	WP	Chi phí	WP
R2	71,08	46	56,777	25	49,904	7
R3	20,452	3	20,395	3	20,395	3
R4	25,597	9	23,893	4	23,893	4
R5	29,374	4	29,183	3	29,183	3
R6	21,92	3	21,898	3	21,898	3
R7	19,531	2	19,531	2	19,531	2
R8	30,899	5	29,906	3	29,906	3
R9	23,315	5	22,493	4	22,493	4
R10	100	52	33,824	4	33,824	4
Tổng số	374,651		287,099		280,226	

Bảng 4.16: Giá trị chi phí và số lượng điểm trung gian, Bộ dữ liệu #A7

Thứ tự robot	Khởi tạo phỏng đoán		ElasticABC		DeZigzagPath	
	Chi phí	WP	Chi phí	WP	Chi phí	WP
R1	52,647	46	34,497	19	26,771	4
R2	27,074	5	26,313	3	26,313	3
R3	29,159	13	23,826	7	23,748	4
R4	41,656	10	38,225	4	38,225	4
R5	31,166	14	25,232	6	25,029	3
R6	23,517	13	20,058	6	19,925	4
R7	27,853	7	26,385	4	26,385	4
R8	55,506	10	54,216	5	54,216	5
R9	43,091	5	42,191	4	42,191	4
R10	36,754	4	36,373	3	36,373	3
Tổng số	368,423		327,316		319,176	



Hình 4.11: Đường đi khi áp dụng kỹ thuật khử gập khúc, TH #4



Hình 4.12: Đường đi khi áp dụng kỹ thuật khử gập khúc, TH #5

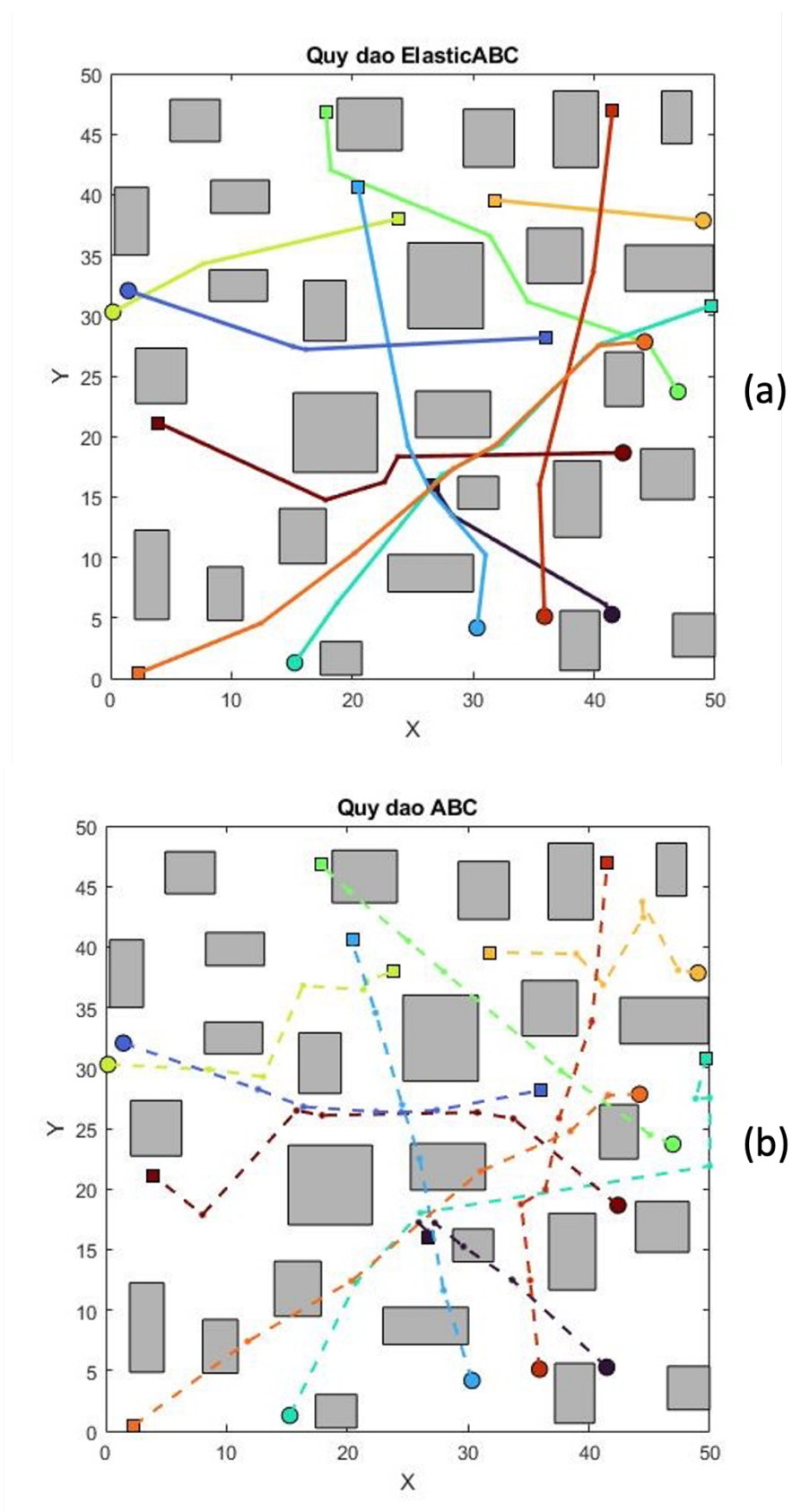
4.4 Đánh giá hoạt động của ElasticABC với ABC truyền thống

Các thử nghiệm ElasticABC đã cho thấy đường đi của robot có vị trí các điểm trung gian thích nghi một cách có hiệu quả với độ phức tạp của môi trường thông qua cơ chế co giãn sử dụng cặp toán tử đối ngẫu kết hợp với khởi tạo phỏng đoán. Thuật toán luôn hoạt động một cách tin cậy trong điều kiện môi trường biến đổi, xem xét vật cản có dạng đa giác lồi. Kết quả đạt được cho thấy sự cân bằng giữa khả năng khám phá và khả năng khai thác. Trên cơ sở đó, ElasticABC được thử nghiệm so sánh với thuật toán ABC truyền thống (cABC), thông số thiết lập được liệt kê như Bảng 4.17. Trong đó, giá trị số lượng điểm trung gian được giữ cố định là 5, số lần thử liên tiếp tăng lên 30; hai thông số còn lại giữ nguyên.

Bảng 4.17: Thông số cho thuật toán ABC truyền thống

Thông số	Giá trị	Mô tả
nPop	25	Số lượng cá thể trong quần thể ong.
iWP	5	Số lượng điểm trung gian
Limit	30	Số lần thử liên tiếp mà ong thợ và ong quan sát thực hiện trước khi từ bỏ một giải pháp
MaxIt	100	Số vòng lặp tối đa của giải thuật ElasticABC

Hình 4.13 minh họa kết quả tìm kiếm đường đi của thuật toán ElasticABC và cABC. Trong đó, đường nét liền thể hiện đường đi tối ưu của ElasticABC và đường nét đứt biểu thị kết quả của cABC. Đường đi của mỗi robot được thể hiện bằng một màu sắc khác biệt. Bảng 4.18 mô tả kết quả chi phí và số lượng điểm trung gian của tương ứng cABC và ElasticABC; mức giảm chi phí (%) và số lượng WP của ElasticABC so với cABC. Số liệu chi phí của phần tử R3, R4, R8 và R9 của thuật toán cABC truyền thống lớn hơn phản ánh tình huống giải pháp được tìm kiếm đã vi phạm hàm phạt va chạm. Giá trị chi phí là kết quả cuối cùng của hàm mục tiêu, tương ứng với độ dài hành trình của từng robot.



Hình 4.13: Kết quả của ElasticABC (a) và ABC truyền thống (b)

Bảng 4.18: Giá trị chi phí và số lượng điểm trung gian

Thứ tự robot	cABC		ElasticABC		Mức giảm	
	Chi phí	WP	Chi phí	WP	% chi phí	WP
R1	26,281	7	17,266	2	34,30	5
R2	29,338	7	25,102	3	14,44	4
R3	1.037,300	7	40,552	6	-	1
R4	1.021,500	7	18,571	4	-	3
R5	54,641	7	46,624	7	14,67	0
R6	43,725	7	42,536	4	2,72	3
R7	46,065	7	41,340	5	10,26	2
R8	1.037,900	7	39,052	5	-	2
R9	1.050,700	7	50,976	7	-	0
R10	35,664	7	35,396	4	0,75	3

Dữ liệu thu được đã thể hiện rõ ràng những lợi thế đáng kể của ElasticABC so với mô hình ABC truyền thống, đặc biệt là về tính khả thi và hiệu quả cấu trúc của hành trình cũng như độ ổn định trong hoạt động.

- Khả năng thoát khỏi bẫy cục bộ đối với phần tử R3, R4, R8 và R9; ElasticABC vẫn tìm được giải pháp đường đi trong khi thuật toán cABC không tìm được.

- Sự cứng nhắc trong cấu trúc đường đi trong thuật toán cABC, số điểm trung gian cố định (7); ngược lại, ElasticABC đã linh hoạt điều chỉnh giảm tùy thuộc vào điều kiện môi trường.

- Hiệu quả hoạt động thể hiện qua mức giảm chi phí cũng như số lượng điểm trung gian thấp hơn rõ rệt giữa ElasticABC so với ABC truyền thống: 10/10 phần tử

có chi phí giảm; 4/10 phần tử có kết quả tìm kiếm, không gặp bất kỳ cục bộ (R3, R4, R8 và R9); 4/10 phần tử có mức giảm hơn 10% (R1, R2, R5 và R7); 2/10 phần tử có mức giảm dưới 5% (R6 và R10).

Như vậy, ElasticABC có khả năng mạnh mẽ trong việc điều hướng và tìm kiếm giải pháp gần tối ưu trong môi trường phức tạp mà mô hình tiêu chuẩn không thể vượt qua. Khoảng cách hiệu suất này bắt nguồn từ một sự thay đổi mô hình cơ bản: bằng cách chuyển từ tối ưu hóa tham số sang tối ưu hóa cấu trúc, ElasticABC điều chỉnh động mật độ điểm trung gian để phù hợp với cấu trúc địa hình môi trường cục bộ, từ đó giải quyết được triệt để hạn chế vốn có của chiến lược tìm kiếm có kích thước cố định.

4.5 Thử nghiệm và đánh giá hoạt động của DCM

Kịch bản mô phỏng được thực hiện để đánh giá hiệu suất của DCM; không gian làm việc được thiết lập bằng một bản đồ số kích thước 25x25; hệ thống đa robot gồm có 50 phần tử. Do những hạn chế về khả năng tính toán của máy tính dùng để mô phỏng, số lượng robot và kích thước bản đồ được giới hạn tương ứng như đã đề cập trong mô hình mô phỏng. Thuật toán tìm đường A* được sử dụng để tạo dữ liệu đầu vào. Từng phần tử được phân công nhiệm vụ di chuyển từ một tọa độ xuất phát được tạo ngẫu nhiên $(q_{10}, q_{20}, \dots, q_{N0})$ đến một tọa độ đích được gán ngẫu nhiên $(q_{1G}, q_{2G}, \dots, q_{NG})$. Yếu tố ngẫu nhiên nói trên làm cho mỗi lần chạy mô phỏng có thể tạo ra một tập dữ liệu các điểm giao cắt đường đi duy nhất. Ngoài ra, sự xuất hiện của các vật cản tĩnh cũng làm biến đổi không gian tìm kiếm, dẫn đến sự thay đổi về vị trí và mật độ của các điểm giao cắt.

Để xem xét hoạt động của DCM trong những bối cảnh môi trường cụ thể, đánh giá dựa trên dữ liệu định lượng từ việc chạy mô hình mô phỏng. Ba trường hợp được đề xuất như sau:

i. Thứ nhất là định lượng sự bất định, DCM được chạy 100 lần trên cùng một tập dữ liệu điểm giao cắt duy nhất. Các chỉ số thống kê mô tả, như giá trị trung bình, độ lệch chuẩn, khoảng tứ phân vị, trung vị, được áp dụng cho giá trị hàm mục tiêu tối

thiếu và số lượng robot mỗi cụ, nhằm đánh giá độ ổn định hiệu suất và sự biến thiên cấu hình của thuật toán.

ii. Thứ hai là phân tích độ nhạy, kiểm chứng DCM khi có biến động đầu vào: bốn tập dữ liệu điểm giao cắt riêng biệt được tạo ra từ các cặp điểm xuất phát-đích ngẫu nhiên khác nhau. DCM được chạy 100 lần trên mỗi tập dữ liệu này để so sánh đặc tính phân cụm, qua đó đánh giá độ nhạy của hệ thống dưới các điều kiện đầu vào thay đổi.

iii. Thứ ba là đánh giá khả năng khái quát của DCM, kiểm chứng qua hai loại môi trường: (1) giảm quy mô xuống 25 robot trong bản đồ không vật cản, và (2) giữ nguyên 50 robot trong môi trường có vật cản ngẫu nhiên. Với 100 lần thực thi cho mỗi trường hợp, kết quả phân tích sẽ chứng minh khả năng ứng dụng rộng rãi của thuật toán trước các thay đổi về quy mô và độ phức tạp môi trường.

Trong cả ba trường hợp, mô hình GMM lựa chọn số lượng thành phần (số cụm) là cố định, $K = 3$; được thúc đẩy bởi sự phân chia không gian làm việc theo trực giác kinh nghiệm. Thuật toán EM được chạy lặp với số vòng lặp tối đa là 100; thuật toán hội tụ thông qua giám sát giá trị khả năng cực đại tích lũy qua mỗi vòng lặp. Mô phỏng dừng khi biến thiên trong giá trị tích lũy giữa các vòng lặp liên tiếp thấp hơn ngưỡng 1×10^{-6} , khi đó ước lượng tham số đã đạt đến ngưỡng ổn định.

Phân tích và đánh giá hiệu suất hoạt động thông qua các thông số bao gồm: giá trị tối thiểu của hàm mục tiêu, f_{objmin} , số lượng phần tử robot trong từng cụm theo mỗi cấu hình và tỷ lệ robot được phân cụm, r_{cl} . Bảng 4.19 tổng hợp kết quả của thông số nói trên và Bảng 4.20 minh họa chỉ số thống kê mô tả đối với dữ liệu f_{objmin} và số lượng robot của các cụm.

Bảng 4.19: Số liệu của trường hợp mô phỏng thứ nhất

f_{objmin}	Số lượng phần tử trong cụm			r_{cl} (%)	f_{objmin}	Số lượng phần tử trong cụm			r_{cl} (%)
	#1	#2	#3			#1	#2	#3	
0,0593	27	25	43	100	0,0564	31	43	25	100
0,0641	25	24	36	100	0,0563	26	36	33	100
0,0565	36	25	35	100	0,0575	37	25	32	100

f_{objmin}	Số lượng phần tử trong cụm			r_{cl} (%)	f_{objmin}	Số lượng phần tử trong cụm			r_{cl} (%)
	#1	#2	#3			#1	#2	#3	
0,0553	36	33	27	100	0,0565	37	25	34	100
0,0635	34	28	23	100	0,0565	34	25	37	100
0,0558	25	33	41	100	0,0519	37	31	33	100
0,0577	29	25	43	100	0,0612	39	35	21	100
0,0565	34	37	25	100	0,0553	33	27	36	100
0,0557	33	35	27	100	0,0614	25	32	29	100
0,0606	23	29	41	100	0,0593	27	43	25	100
0,0575	32	37	25	100	0,0593	27	43	25	100
0,0565	37	25	34	100	0,0544	34	28	35	100
0,0565	36	35	25	100	0,0570	25	30	43	100
0,0565	36	25	35	100	0,0525	37	30	33	100
0,0558	41	25	33	100	0,0565	37	34	25	100
0,0553	33	27	36	100	0,0541	33	35	29	100
0,0593	25	27	43	100	0,0584	34	39	23	100
0,0565	25	34	37	100	0,0531	29	35	35	100
0,0558	33	25	41	100	0,0538	31	35	31	100
0,0563	33	36	26	100	0,0595	30	24	38	100
0,0531	35	29	35	100	0,0584	43	28	25	100
0,0531	29	35	35	100	0,0561	32	42	25	100
0,0586	23	33	40	100	0,0565	25	37	34	100
0,0565	34	25	37	100	0,0577	29	43	25	100
0,0563	33	26	36	100	0,0558	25	32	43	100
0,0561	25	42	32	100	0,0586	40	33	23	100
0,0558	43	32	25	100	0,0563	33	26	36	100
0,0525	32	33	34	100	0,0558	32	43	25	100
0,0521	35	32	33	100	0,0593	27	25	43	100
0,0565	35	36	25	100	0,0553	34	35	27	100
0,0565	37	34	25	100	0,0577	29	43	25	100
0,0538	31	35	31	100	0,0570	30	43	25	100
0,0570	25	43	30	100	0,0565	25	36	35	100
0,0563	36	33	26	100	0,0558	25	32	43	100
0,0536	34	35	29	100	0,0628	25	34	26	100
0,0564	31	43	25	100	0,0565	36	25	35	100
0,0541	29	35	33	100	0,0549	28	33	35	100
0,0538	31	35	31	100	0,0565	25	34	37	100
0,0521	35	33	32	100	0,0538	35	31	31	100
0,0614	29	25	32	100	0,0541	33	35	29	100
0,0545	36	28	33	100	0,0552	29	35	31	100
0,0539	30	32	35	100	0,0589	38	31	24	100
0,0614	29	25	32	100	0,0553	33	27	36	100

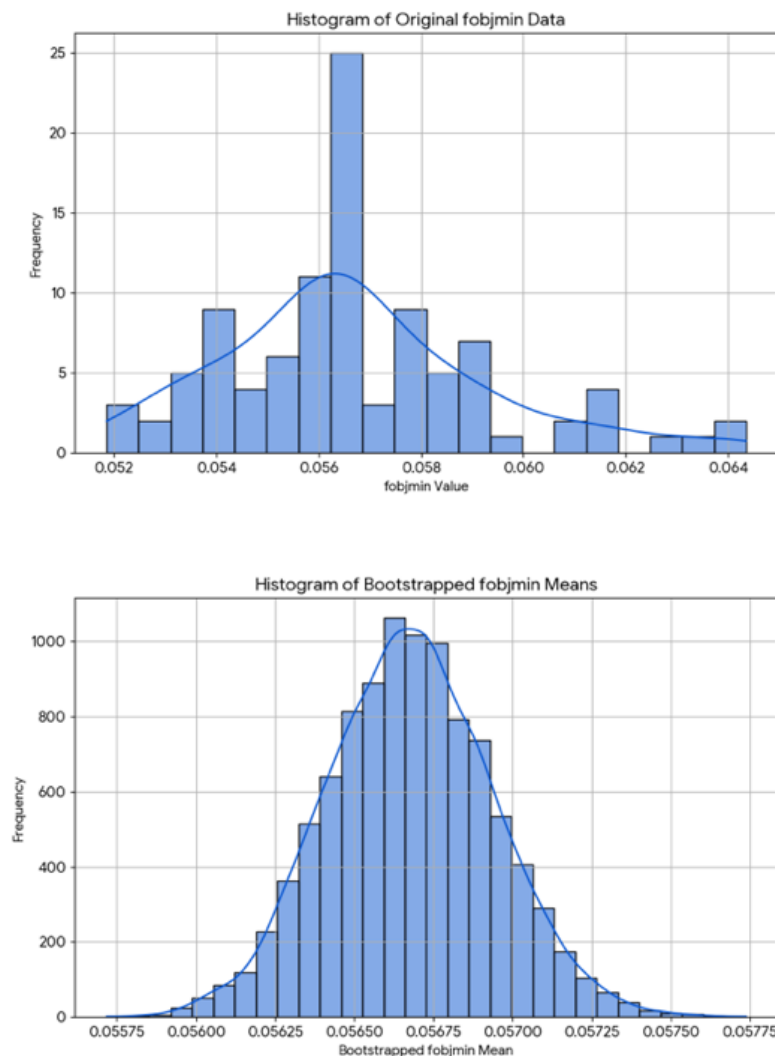
f_{objmin}	Số lượng phần tử trong cụm			r_{cl} (%)	f_{objmin}	Số lượng phần tử trong cụm			r_{cl} (%)
	#1	#2	#3			#1	#2	#3	
0,0565	37	34	25	100	0,0538	31	35	31	100
0,0531	35	35	29	100	0,0579	36	38	23	100
0,0586	23	40	33	100	0,0579	36	23	38	100
0,0579	36	23	38	100	0,0565	25	37	34	100
0,0593	25	43	27	100	0,0644	23	36	26	100
0,0558	25	41	33	100	0,0614	29	32	25	100
0,0544	28	34	35	100	0,0577	25	29	43	100

Bảng 4.20: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D1

	f_{objmin}	Số lượng phần tử trong cụm		
		#1	#2	#3
Bộ dữ liệu #D1				
Trung bình	0,0567	31,38	32,67	31,82
Độ lệch chuẩn	0,0026	4,89	5,73	5,9
Trung vị	0,0565	32	33	32,5
Tứ phân vị thứ nhất (Q1)	0,0553	27	27,75	25,75
Tứ phân vị thứ ba (Q3)	0,0579	35	35,25	35,25
Khoảng tứ phân vị (IQR)	0,0026	8	7,5	9,5
Khoảng tin cậy 95% (2σ)	(0,0515 0,0619)			

Chỉ số thống kê của f_{objmin} cho thấy giá trị của hàm tối ưu đạt được sự ổn định và nhất quán rất cao. Dữ liệu có xu hướng tập trung, thể hiện thông qua sự hội tụ giữa giá trị trung bình 0,0567 và trung vị 0,0565; đồng thời, mức chênh lệch 0,0003 là không đáng kể so với độ lệch chuẩn 0,0026. Sự chênh lệch này cũng phản ánh phân phối dữ liệu đối xứng với độ lệch về bên phải không đáng kể. Hơn nữa, một biểu hiện rõ ràng về mức độ biến thiên thấp của dữ liệu chính là độ lệch chuẩn 0,0026 và khoảng tứ phân vị 0,0026 rất nhỏ, phản ánh mật độ tập trung dữ liệu rất lớn quanh giá trị trung bình, Hình 4.14 minh họa biểu đồ phân phối dữ liệu. Sự tương đồng giữa giá trị khoảng tin cậy 95% (2 σ) là (0,0515 0,0619) và khoảng thực nghiệm quan sát được của dữ liệu, từ 0,0518 đến 0,0644 củng cố thêm sự nhất quán nói trên. Kỹ thuật bootstrapping đã được áp dụng với 10.000 mẫu tái lập để đánh giá thêm về độ tin cậy

của giá trị trung bình f_{objmin} , kỹ thuật, minh họa như biểu đồ của Hình 4.14. Biểu đồ dữ liệu tuân theo quy luật phân phối chuẩn và có mật độ tập trung rất cao quanh giá trị trung bình 0,0567, trùng khớp với trung bình mẫu gốc, cho thấy đây là một ước lượng chính xác và không chệch. Độ lệch chuẩn của phân phối tái mẫu này có giá trị cực nhỏ là 0,00026, thấp hơn đáng kể so với độ lệch chuẩn của dữ liệu gốc (0,0026), phản ánh độ phân tán của các điểm dữ liệu riêng lẻ. Ngoài ra, khoảng tin cậy 95% (0,0561 0,0572) rất hẹp, thu được từ phân tích bootstrap, càng khẳng định nhận định trên.



Hình 4.14: Biểu đồ tần suất dữ liệu f_{objmin} và giá trị tái lập

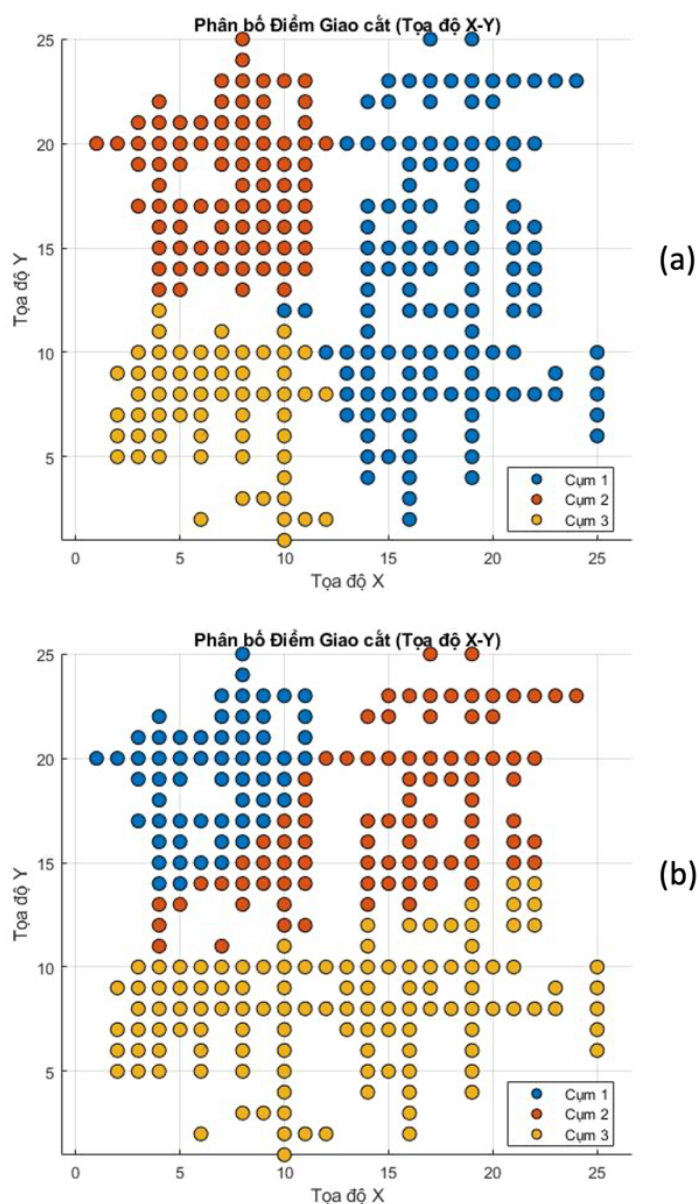
Các chỉ số thống kê đã cho thấy xu hướng tập trung tương đồng trong phân chia phần tử robot giữa ba cụm. Cụ thể, giá trị trung bình và trung vị của tất cả các cụm đều tập trung trong một khoảng hẹp từ 31 đến 33 robot; sự gần nhau giữa hai chỉ số này trong mỗi cụm gợi ý rằng dữ liệu có phân bố tương đối đối xứng. Mặc dù xu hướng tập trung là tương đồng, mức độ phân tán giữa các cụm có sự khác biệt nhất định. Cụ thể, Cụm số 3 thể hiện độ biến thiên cao hơn (độ lệch chuẩn 5,90 và khoảng tứ phân vị là 9,50) so với tính ổn định cao của Cụm số 1 và Cụm số 2. Điều này chỉ ra rằng số lượng robot trong Cụm số 3 phân tán nhiều hơn so với hai cụm còn lại. Tuy vậy, nhìn chung sự phân bố robot vẫn đảm bảo sự cân bằng cần thiết.

Xét một cấu hình phân cụm cụ thể đã tìm được (35 25 24), Bảng 4.21 liệt kê các phần tử robot theo từng cụm, tương ứng với Hình 4.15.a minh họa đồ thị điểm giao cắt; Hình 4.15.b là đồ thị minh của một cấu hình khác cho cùng một bộ dữ liệu.

Bảng 4.21: Phân bố robot theo cụm

	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
Cụm 1	x	x		x			x	x	x	x
Cụm 2	x	x	x		x	x			x	
Cụm 3		x								x
	R11	R12	R13	R14	R15	R16	R17	R18	R19	R20
Cụm 1	x	x			x			x		
Cụm 2	x	x	x	x		x			x	x
Cụm 3	x		x				x	x	x	x
	R21	R22	R23	R24	R25	R26	R27	R28	R29	R30
Cụm 1	x	x		x		x		x	x	x
Cụm 2			x	x	x	x	x			
Cụm 3			x	x	x			x		x
	R31	R32	R33	R34	R35	R36	R37	R38	R39	R40
Cụm 1	x	x	x	x	x	x	x	x		
Cụm 2	x					x		x		x

Cụm 3		x				x		x	x	
	R41	R42	R43	R44	R45	R46	R47	R48	R49	R50
Cụm 1	x	x	x	x	x	x	x	x	x	
Cụm 2		x							x	x
Cụm 3	x		x	x	x		x	x		x



Hình 4.15: Phân bố điểm giao cắt theo cụm

Kết quả mô phỏng đã làm rõ độ tin cậy về hiệu suất phân cụm của DCM khi đối mặt với sự biến thiên của cấu hình cụm. Hiệu suất tổng thể, được định lượng bằng các giá trị f_{objmin} , thể hiện sự ổn định. Kết luận này được củng cố bởi khoảng tin cậy 95% rất hẹp (0,0561 0,0572) thu được thông qua kỹ thuật bootstrapping. Sự nhất quán này đặc biệt đáng chú ý khi đối chiếu với những biến động khác nhau trong các cấu hình cụm cơ sở, nơi số lượng robot mỗi cụm thay đổi giữa các lần chạy mô phỏng khác nhau. Khả năng duy trì mức độ ổn định và nhất quán, ngay cả khi phân bố robot thay đổi, đã chứng minh độ tin cậy về hiệu suất của phương pháp DCM. Phương pháp này thích ứng hiệu quả với các biến động nội bộ mà không làm ảnh hưởng đến mục tiêu hiệu suất tổng thể.

Bảng 4.22: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D2 đến #D5

	Bộ mẫu tái lập	f_{objmin}	Số lượng phần tử trong cụm		
			#1	#2	#3
Bộ dữ liệu #D2					
Trung bình	0,0642	0,0642	29,81	30,28	29,75
Độ lệch chuẩn	0,00021	0,0021	7,65	7,88	8,21
Trung vị	0,0642	0,0653	29	29	29
Tứ phân vị thứ nhất (Q1)	0,0640	0,0646	23	23	22
Tứ phân vị thứ ba (Q3)	0,0643	0,0655	36,25	38	38
Khoảng tứ phân vị (IQR)	0,00028	0,0009	13,25	15	16
Khoảng tin cậy 95% (2σ)	(0,0638, 0,0646)	(0,0599, 0,0685)			
Bộ dữ liệu #D3					
Trung bình	0,0611	0,0611	29,96	29,46	31,29
Độ lệch chuẩn	0,00014	0,0014	6,89	6,13	7,18
Trung vị	0,0611	0,0615	28	27	28
Tứ phân vị thứ nhất (Q1)	0,0610	0,0594	24	24	25,5
Tứ phân vị thứ ba (Q3)	0,0612	0,0623	37,25	37	38
Khoảng tứ phân vị (IQR)	0,00018	0,0029	13,25	13	12,5
Khoảng tin cậy 95% (2σ)	(0,0608, 0,0613)	(0,0583, 0,0638)			
Bộ dữ liệu #D4					
Trung bình	0,0563	0,0563	32,15	32,13	31,27
Độ lệch chuẩn	0,00017	0,0017	5,12	5,33	4,94

	Bộ mẫu tái lập	f_{objmin}	Số lượng phần tử trong cụm		
			#1	#2	#3
Trung vị	0,0563	0,0566	29	29	29
Tứ phân vị thứ nhất (Q1)	0,0562	0,0564	27	27	27
Tứ phân vị thứ ba (Q3)	0,0564	0,0570	37	37	36
Khoảng tứ phân vị (IQR)	0,00022	0,0006	10	10	9
Khoảng tin cậy 95% (2 σ)	(0,0560, 0,0566)	(0,0529, 0,0596)			
<i>Bộ dữ liệu #D5</i>					
Trung bình	0,0645	0,0645	30,57	29,3	27,69
Độ lệch chuẩn	0,0003	0,0031	5,99	6,08	6,32
Trung vị	0,0645	0,0636	34	32	28
Tứ phân vị thứ nhất (Q1)	0,0643	0,0625	25,5	24	22
Tứ phân vị thứ ba (Q3)	0,0647	0,0667	35	34	34
Khoảng tứ phân vị (IQR)	0,0004	0,0042	9,5	10	12
Khoảng tin cậy 95% (2 σ)	(0,0639, 0,0651)	(0,0584, 0,0706)			

Các chỉ số thống kê phục vụ phân tích độ nhạy được tổng hợp trong Bảng 4.22; bao gồm giá trị f_{objmin} , số lượng phần tử robot trên mỗi cụm, bộ mẫu tái lập 10.000 giá trị của kỹ thuật bootstrapping, áp dụng lên bộ dữ liệu f_{objmin} gốc. Nhìn chung, số liệu cho thấy DCM có khả năng thích ứng cao: (1) duy trì sự ổn định nội tại trong từng bộ dữ liệu (độ lệch chuẩn thấp); (2) linh hoạt thay đổi cấu hình phân cụm (giá trị trung bình thay đổi) để phù hợp với đặc thù của từng tập dữ liệu đầu vào mới. Điều này chứng minh thuật toán không bị "rập khuôn" mà thực sự biến đổi theo dữ liệu đầu vào.

Việc phân tích từng bộ dữ liệu riêng lẻ, từ Bộ dữ liệu #D2 đến Bộ dữ liệu #D5, tiếp tục cho thấy DCM mức hiệu suất ổn định và nhất quán của DCM; điều này tương tự kết quả quan sát được trong kịch bản mô phỏng đầu tiên. Xét trên tổng thể cả năm bộ dữ liệu, phân tích độ nhạy lại xác nhận một thể mạnh cốt lõi khác của DCM: sự phụ thuộc chặt chẽ vào từng bộ dữ liệu cụ thể. Điều này được minh chứng định lượng qua cả hàm mục tiêu và sự phân bố robot. Giá trị trung bình và trung vị của f_{objmin}

dao động đáng kể, từ mức thấp nhất là 0,0563 ở Bộ dữ liệu #D4 đến mức cao nhất là 0,0645 ở Bộ dữ liệu #D5. Điều này được củng cố thêm bởi chỉ số liên quan đến phân bố số lượng phần tử robot theo cụm. Trong đó, giá trị trung bình và độ lệch chuẩn của mỗi cụm thay đổi đáng kể qua các bộ dữ liệu, phản ánh sự thành công của DCM trong việc tái phân bố robot một cách linh hoạt để tương thích với tập dữ liệu đầu vào cụ thể. Như vậy, phân tích độ nhạy đã xác nhận sự thích nghi của DCM mà không bị ràng buộc bởi một cấu trúc phân cụm tối ưu duy nhất cho mọi trường hợp. DCM có khả năng duy trì sự ổn định cho một bài toán cụ thể ngay cả khi có những thay đổi trong cấu hình nội bộ của từng cụm; đồng thời có thể tái lập hoàn toàn cấu hình cụm để phù hợp với các bài toán mới.

Bảng 4.23: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D6 và #D7

	Bộ mẫu tái lập	f_{objmin}	Số lượng phần tử trong cụm		
			#1	#2	#3
Bộ dữ liệu #D6					
Trung bình	0,1413	0,1413	13,24	13,34	13,82
Độ lệch chuẩn	0,0011	0,0107	3,3	3,2	3,14
Trung vị	0,1413	0,1355	14	14	15
Tứ phân vị thứ nhất (Q1)	0,1406	0,1355	10	10	11
Tứ phân vị thứ ba (Q3)	0,142	0,15	15	16	16
Khoảng tứ phân vị (IQR)	0,0014	0,0145	5	6	5
Khoảng tin cậy 95% (2σ)	(0,1391, 0,1434)	(0,1200, 0,1626)			
Bộ dữ liệu #D7					
Trung bình	0,0626	0,0626	30,76	30,14	31,45
Độ lệch chuẩn	0,00023	0,0023	8,52	8,25	8,41
Trung vị	0,0625	0,0623	27,5	26	30
Tứ phân vị thứ nhất (Q1)	0,0624	0,0619	23	23	23
Tứ phân vị thứ ba (Q3)	0,0627	0,0625	38	37,25	37,25
Khoảng tứ phân vị (IQR)	0,00031	0,00069	15	14,25	14,25
Khoảng tin cậy 95% (2σ)	(0,06210, 0,06301)	(0,05799, 0,06712)			

Các chỉ số thống kê được trình bày trong Bảng 4.23 được dùng kiểm chứng khả năng tổng quát hóa của DCM; là số liệu tương ứng với hai trường hợp của kịch bản mô phỏng thứ ba. Đó là trường hợp giảm số lượng robot, với 25 robot hoạt động trong bản đồ 25x25 không có vật cản, và trường hợp sử dụng đầy đủ 50 robot trong bản đồ 25x25 có vật cản. Kỹ thuật bootstrapping được áp dụng cho dữ liệu f_{objmin} gốc với 10.000 mẫu tái lập, để ước lượng độ ổn định của giá trị trung bình của f_{objmin} . Dù vận hành ở quy mô nhỏ đơn giản (25 robot, không vật cản) hay quy mô lớn phức tạp (50 robot, có vật cản), các chỉ số thống kê và kết quả bootstrapping đều cho thấy sự ổn định tương đương. Điều này khẳng định tính chất độc lập với môi trường của giải pháp DCM.

Phân tích chỉ số thống kê của Bộ dữ liệu #D6 và Bộ dữ liệu #D7 đã cho thấy hiệu suất phân cụm ổn định và tin cậy qua hai trường hợp vận hành khác nhau. Kết luận này được củng cố thông qua kết quả thống kê cho bộ dữ liệu tái mẫu. Ví dụ, trong Bộ dữ liệu #D7, trung bình bootstrapping là 0,0626, đồng nhất với trung bình mẫu gốc; sai số chuẩn tương ứng là 0,00023, nhỏ hơn đáng kể so với độ lệch chuẩn của dữ liệu gốc là 0,0023. Khoảng tin cậy 95% hẹp (0,06210 0,06301) đã củng cố tính hợp lệ và độ ổn định của ước lượng trung bình. Hơn nữa, cơ chế phân bố phần tử theo cụm đã chỉ ra được xu hướng cân bằng rõ ràng trong cả hai trường hợp. Các cụm thể hiện sự tương đồng về xu hướng tập trung và độ phân tán; phản ánh phương pháp phân cụm không bị thiên lệch và biến động; việc gán robot là đồng đều qua các lần chạy thực nghiệm. Quan trọng hơn, các phân tích này cho thấy sự ổn định tương đương trong hiệu suất phân cụm của DCM dưới hai trường hợp riêng biệt: giảm số lượng tác tử với 25 robot trong bản đồ không vật cản, và đầy đủ 50 robot trong bản đồ có vật cản. Khi xem xét kết hợp với các đánh giá cho kịch bản đầu tiên, điều này đã chứng minh bằng thực nghiệm khả năng tổng quát hóa và tính chất độc lập của phương pháp DCM với môi trường hoạt động của hệ thống đa robot.

Các phân tích định lượng đã khẳng định hiệu quả phân cụm của phương pháp DCM; đặc biệt là khả năng duy trì cân bằng tải tổng thể thông qua việc luôn giữ ổn định giá trị hàm mục tiêu f_{obj} . Tuy nhiên, một quan sát đáng chú ý là độ lệch chuẩn

về số lượng phân tử trong các cụm lại biến động lớn, thể hiện nhất quán qua hệ số biến thiên (CV) dao động từ 15% đến 30% trong mọi kịch bản thử nghiệm. Dưới góc độ hình học, sự biến động số liệu này minh chứng cho sự tồn tại của các vùng giao thoa tương đối lớn giữa các cụm, phản ánh đúng bản chất xác suất của mô hình GMM. Thay vì là sự thiếu ổn định, đặc tính này biểu hiện cho tính linh hoạt và khả năng thích nghi nội tại của DCM trước sự phức tạp của môi trường. Để làm rõ cơ chế thích nghi này, phần tiếp theo sẽ tập trung phân tích và đánh giá sâu hơn về độ phân tán không gian của các cụm.

Các chỉ số thống kê được trình bày trong Bảng 4.24 được dùng kiểm chứng sự sắp xếp không gian và tính nhất quán của mỗi cụm thông qua đánh giá định lượng xu hướng tập trung và độ phân tán của dữ liệu. Các chỉ số bao gồm giá trị trung bình, độ lệch chuẩn (SD), trung vị và khoảng tứ phân vị (IQR) cho dữ liệu tọa độ X và Y của ba tâm cụm riêng biệt trong kịch bản mô phỏng đầu tiên.

Kết quả về độ lệch chuẩn trên tất cả các cụm khá tương đồng, dao động trong phạm vi hẹp từ 3,84 đến 4,47. Cụ thể, Cụm #2 có sự biến động không gian lớn nhất, minh chứng bởi tọa độ Y có độ lệch chuẩn cao nhất (4,47) và IQR rộng nhất (9,43). Tương tự, tọa độ X của cụm này cũng có độ phân tán cao với SD là 4,06 và IQR là 7,88. Cụm #3 cho thấy độ tập trung cao hơn, với tọa độ X hiển thị độ lệch chuẩn thấp nhất (3,84) và IQR nhỏ nhất (7,41). Tọa độ Y của cụm này có độ trải rộng hơn một chút nhưng vẫn thấp hơn Cụm #2. Trên toàn bộ tập dữ liệu, giá trị độ lệch chuẩn luôn nhỏ hơn IQR. Điều này chỉ ra rằng phân bố không gian không hoàn toàn đối xứng và chịu ảnh hưởng bởi các giá trị ngoại lai, làm gia tăng độ trải của dữ liệu.

Bảng 4.24: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D8

		Trung bình	Độ lệch chuẩn	Trung vị	Khoảng tứ phân vị (IQR)
<i>Bộ dữ liệu #D8</i>					
Cụm #1	X	11,2104	4,1715	12,1503	7,8841
	Y	13,7347	4,3226	13,4684	8,0918
Cụm #2	X	10,8981	4,0555	11,8624	7,8761
	Y	13,7915	4,4685	13,2703	9,4265
Cụm #3	X	11,5118	3,8358	12,2046	7,4063
	Y	14,0077	4,1988	13,4684	7,9833

Hình dáng phân bố của cụm được bộc lộ thông qua so sánh giá trị trung bình và trung vị giúp làm rõ độ lệch. Đối với tọa độ X, giá trị trung bình luôn thấp hơn trung vị trên cả ba cụm, gợi ý một độ lệch âm do sự xuất hiện của các điểm giao cắt ở tọa độ thấp. Ngược lại, đối với tọa độ Y, giá trị trung bình lại cao hơn trung vị, ngụ ý độ lệch dương nhẹ. Tuy nhiên, biên độ chênh lệch giữa hai thước đo này tương đối nhỏ, cho thấy tác động của các điểm ngoại lai lên cấu trúc tổng thể là hữu hạn và nằm trong ngưỡng kiểm soát của thuật toán.

Quan trọng hơn, sự kết hợp các chỉ số độ phân tán nói trên với tọa độ tâm cụm đã chỉ ra đặc điểm hình thái học quan trọng của các cụm; đó là sự giao thoa không gian. Cụ thể, tọa độ trung bình của cả ba tâm cụm đều hội tụ về một khu vực rất hẹp, giá trị X trong khoảng (11,0 11,5), giá trị Y trong khoảng (13,7 14,0); độ lệch chuẩn được duy trì ở mức cao, trong khoảng (4,0 4,5). Vùng bao phủ của mỗi cụm đều trải rộng trên một diện tích lớn của bản đồ và chúng không tách biệt mà hình thành một cấu trúc chồng lấn. Trong cấu trúc này, ranh giới giữa các cụm là mềm; một robot di chuyển trong vùng này sẽ chịu ảnh hưởng xác suất gần như ngang nhau từ cả ba cụm. Số liệu định lượng khẳng định sự tồn tại của vùng giao thoa này và sự biến động về số lượng phần tử phân chia cho từng cụm trong các kịch bản thử nghiệm. Thay vì cố định robot vào một nhóm cứng nhắc, DCM tận dụng sự chồng lấn này để linh hoạt điều chuyển robot giữa các nhóm, hình thành cơ chế thích nghi nội tại của DCM: khả năng "co giãn" không gian và chia sẻ tải trọng linh hoạt để đạt được yêu cầu cân bằng tải của toàn bộ hệ thống.

Bảng 4.25: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D9 và #D10

		Trung bình	Độ lệch chuẩn	Trung vị	Khoảng tứ phân vị (IQR)
<i>Bộ dữ liệu #D9</i>					
Cụm #1	X	12,9062	4,3450	13,7079	4,3731
	Y	11,7953	5,0474	12,3247	10,4779
Cụm #2	X	11,4155	5,0107	12,5457	11,4618
	Y	12,182	4,6467	12,3745	8,2482
Cụm #3	X	11,5598	5,0451	12,4675	11,539
	Y	11,9362	4,5711	12,2486	8,2493
<i>Bộ dữ liệu #D10</i>					

Cụm #1	X	13,2477	3,6791	13,4516	4,5116
	Y	10,8499	4,6917	10,4646	8,5004
Cụm #2	X	13,4294	3,4385	13,4713	3,0461
	Y	9,9421	4,5823	8,0729	8,8684
Cụm #3	X	12,0134	3,4473	13,4352	5,9579
	Y	11,7233	4,9798	10,6778	11,2516

Bảng 4.25 tổng hợp các chỉ số thống kê cho tọa độ X và Y của ba tâm cụm riêng biệt trong kịch bản mô phỏng thứ hai với hai bộ dữ liệu khác nhau; được dùng để mở rộng các phân tích tương tự trong những kịch bản thử nghiệm khác với dữ liệu hoàn toàn khác. Kết quả thống kê từ hai bộ dữ liệu này cung cấp cái nhìn đa chiều hơn về hành vi không gian của các cụm.

Phân tích mở rộng trên hai bộ dữ liệu bổ sung cho thấy độ phân tán của các cụm có sự nhạy cảm đáng kể đối với đặc thù dữ liệu đầu vào, đặc biệt qua chỉ số khoảng tứ phân vị (IQR). Ví dụ, trong khi IQR duy trì ở mức trung bình ở các kịch bản trước, nó lại mở rộng thành một phân phối rất rộng trong Bộ dữ liệu #D9 (IQR lên tới 11,46), và sau đó đột ngột thu hẹp thành một trạng thái tập trung cao độ trong Bộ dữ liệu #D10 (IQR của Cụm #2 chỉ còn 3,05). Sự thay đổi này phản ánh trực tiếp bản chất ngẫu nhiên của các cặp điểm xuất phát/đích; kịch bản #10 đã vô tình tạo ra một luồng giao thông "nút cổ chai" tại một khu vực hẹp, và thuật toán DCM đã phản ứng chính xác bằng cách thu hẹp vùng lõi cụm, tương ứng với 50% dữ liệu trung tâm, để bao phủ mật độ cao này. Ngược lại, độ lệch chuẩn lại duy trì sự ổn định tương đối, dao động trong khoảng (3,5 5,0). Mặc dù hình dáng vùng lõi, căn cứ theo IQR thay đổi linh hoạt để khớp với dữ liệu, nhưng độ phủ tổng thể của cụm vẫn luôn duy trì một phạm vi ảnh hưởng đủ rộng để đảm bảo tính kết nối.

Tác động của các điểm ngoại lai và độ lệch cũng cung cấp những minh chứng rõ nét về khả năng thích nghi hình học của DCM. Trong Bộ dữ liệu #D8, mô hình khá nhất quán với lệch âm ở X và lệch dương ở Y; quy luật này đã bị phá vỡ hoàn toàn trong Bộ dữ liệu #D9, nơi hầu hết các tọa độ đều biểu hiện độ lệch âm. Sự thay đổi đột ngột về hình dạng phân phối này chứng minh rằng các hình thái của các cụm không bị cố định mà chúng mang thuộc tính động, tự động biến dạng để vươn tới các điểm ngoại lai xuất hiện ngẫu nhiên trong từng kịch bản.

Bên cạnh các chỉ số về độ phân tán, dữ liệu thử nghiệm cho thấy tâm cụm không hề cố định mà thay đổi theo kịch bản thử nghiệm. Cụ thể, kết quả của Bộ dữ liệu #D9 thể hiện xu hướng hội tụ trong một khoảng tọa độ rất hẹp tương tự như Bộ dữ liệu #D8; mức chênh lệch không lớn giữa các tâm cụm. Ngược lại, Bộ dữ liệu #D10 cho thấy những kết quả khác biệt. Tâm của Cụm #2 đã dịch chuyển rõ rệt xuống phía dưới trục Y (giá trị trung bình $Y \approx 9,94$), tách biệt đáng kể so với Cụm #3 ($Y \approx 11,72$) và Cụm #1 ($Y \approx 10,85$). Điều này phản ánh sự thay đổi trong phân bố điểm giao cắt đã tác động đến cấu trúc của các cụm.

Như vậy, kết hợp nhận định từ Bộ dữ liệu #D9 và #D10 với kết luận từ Bộ dữ liệu #D8, cơ chế thích nghi nội tại của DCM về không gian dựa trên hai nguyên tắc:

- Thứ nhất là khả năng chống lẫn linh hoạt, ngay cả trong trường hợp dữ liệu tập trung cục bộ như Bộ dữ liệu #D10, độ lệch chuẩn vẫn duy trì ở mức đủ lớn để tạo ra các vùng giao thoa. Điều này đảm bảo rằng dù "lõi" cụm có thu hẹp lại, "vỏ" cụm vẫn vươn ra để duy trì liên kết với các cụm lân cận, ngăn chặn sự đứt gãy trong cấu trúc không gian.

- Thứ hai là khả năng co giãn không gian, DCM không áp đặt cấu trúc lên dữ liệu, mà ngược lại để dữ liệu định hình cấu trúc. Đây chính là yếu tố cốt lõi làm cho thuật toán duy trì hiệu suất phân cụm ổn định bất chấp sự hỗn loạn của đầu vào. Sự biến động của tọa độ tâm chính là minh chứng rõ nét nhất cho tính chất hướng dữ liệu của DCM. Theo đó, vị trí của các cụm hoàn toàn được quyết định bởi tình huống thực tế.

Tóm lại, phân tích chuyên sâu về độ nhạy và hình thái không gian qua các bộ dữ liệu đã làm sáng tỏ bản chất động của phương pháp DCM. Sự biến động đáng kể của các chỉ số IQR, độ lệch chuẩn, giá trị trung bình không phải là dấu hiệu của sự bất ổn, mà trái lại, là minh chứng rõ ràng cho năng lực thích ứng. Chính cơ chế thích nghi không gian linh hoạt này là nền tảng giúp hệ thống luôn duy trì được sự cân bằng tải trọng và hiệu suất phân cụm tối ưu, khẳng định tính ưu việt của phương pháp tiếp cận phân cụm mềm trong môi trường đa robot phức tạp.

Phân tích và đánh giá được mở rộng sang kịch bản thứ ba để xem xét khả năng tổng quát hoá của phương pháp, xét trong hai trường hợp; thứ nhất là môi trường không vật cản với số lượng phần tử giảm xuống 25; thứ hai là môi trường có vật cản với số lượng phần tử giữ nguyên 50. Bộ dữ liệu #D11 và Bộ dữ liệu #D12, minh hoạ trong Bảng 4.26, là chỉ số thống kê tương ứng hai trường hợp, bao gồm giá trị trung bình, độ lệch chuẩn, trung vị và khoảng tứ phân vị (IQR) cho ba tâm cụm riêng biệt.

Bảng 4.26: Chỉ số thống kê của dữ liệu, Bộ dữ liệu #D11 và #D12

		Trung bình	Độ lệch chuẩn	Trung vị	Khoảng tứ phân vị (IQR)
<i>Bộ dữ liệu #D11</i>					
Cụm #1	X	11,3759	3,1236	10,4987	6,5131
	Y	10,5559	2,0367	10,7478	4,2595
Cụm #2	X	11,0005	3,3615	10,4987	7,5278
	Y	10,6413	1,8948	10,7478	4,2595
Cụm #3	X	9,065	2,81	6,9854	3,515
	Y	10,4636	1,7834	10,7478	3,0242
<i>Bộ dữ liệu #D12</i>					
Cụm #1	X	13,908	4,0592	11,5057	7,0225
	Y	11,7269	5,1565	11,4828	7,6165
Cụm #2	X	14,0918	3,9118	11,4931	6,873
	Y	11,5559	5,0124	11,4932	7,6959
Cụm #3	X	13,3515	3,7813	11,4369	6,6379
	Y	11,8418	5,1516	11,4347	7,6009

Trong Bộ dữ liệu #D11 tương ứng với điều kiện không gian trống và mật độ robot thấp, các tọa độ Y duy trì độ tập trung cao và tính đối xứng với sự tương đồng giữa giá trị trung bình và trung vị (độ lệch chuẩn trong khoảng 1,78 đến 2,03) thì các tọa độ X lại phân tán lớn hơn (độ lệch chuẩn trong khoảng 2,81 đến 3,36) và lệch phải. Hiệu ứng này biểu hiện rõ rệt nhất tại Cụm #3, nơi giá trị trung bình của X (9,065) lớn hơn đáng kể so với trung vị (6,985). Sự chênh lệch này chỉ ra rằng, tồn tại một nhóm nhỏ các điểm giao cắt ngoại lai có giá trị cao theo trục X.

Ngược lại, khi chuyển sang môi trường phức tạp chứa vật cản, tương ứng Bộ dữ liệu #D12, độ phân tán dữ liệu tăng vọt đồng loạt trên cả hai trục tọa độ, phản ánh qua việc độ lệch chuẩn và IQR đều tăng lên đáng kể ở tất cả các cụm. Đặc biệt, một xu hướng nhất quán đã xuất hiện: giá trị trung bình cao hơn đáng kể so với trung vị

trên mọi chiều dữ liệu, chỉ ra sự lệch phải phổ biến. Hiện tượng này phản ánh tác động của môi trường, robot phải di chuyển theo các quỹ đạo phức tạp hơn để tránh vật cản, xuất hiện nhiều hơn các điểm giao cắt nằm xa tâm.

Các kết quả định lượng đã cung cấp minh chứng rõ ràng về khả năng tổng quát hoá của phương pháp DCM: sự thích nghi cao đối với các điều kiện môi trường thay đổi. Phương pháp sở hữu cơ chế điều chỉnh động hành vi phân cụm để phản ánh chính xác các ràng buộc vật lý, khẳng định tính linh hoạt và khả năng ứng dụng rộng rãi của thuật toán trong những tình huống vận hành đa dạng mà không cần tinh chỉnh thủ công các tham số hình học.

4.6 Thảo luận về thuật toán ElasticABC và DCM

ElasticABC

ElasticABC là biến thể của ABC, tập trung cải tiến yếu tố nội tại của thuật toán ABC nguyên bản; đó là tối ưu cấu trúc sử dụng cặp toán tử đối ngẫu Co và Dẫn trong giai đoạn tìm kiếm của ong thợ, chiến lược khởi tạo phỏng đoán ở giai đoạn khởi tạo quần thể. Những đóng góp này góp phần hình thành tính mới của ElasticABC, đó là số lượng điểm trung gian thích nghi theo điều kiện của không gian làm việc và quỹ đạo cho từng phần tử robot, khác biệt với cơ chế thiết lập số lượng điểm trung gian cố định của ABC nguyên bản. Sự khác biệt của ElasticABC so với những biến thể khác đã được công bố gần đây về thành phần cặp toán tử đối ngẫu Co và Dẫn được mô tả trong Bảng 4.27; về kỹ thuật DeZigzagPath được mô tả trong Bảng 4.28.

Bảng 4.27: Đối chiếu đóng góp khoa học của cặp toán tử đối ngẫu Co và Dẫn

Biến thể	Mô tả
ElasticABC	Tích hợp cặp toán tử đối ngẫu Co và Dẫn trong giai đoạn tìm kiếm của ong thợ; khởi tạo phỏng đoán sử dụng toán tử Dẫn.
Co-GLABC [71]	Sử dụng khung đồng tiến hóa cho hai nhóm độc lập thuộc quần thể, tích hợp thêm thông tin của giải pháp tốt nhất toàn cục trong phương trình tìm kiếm.

Biến thể	Mô tả
ADL-ABC [74]	Cơ chế giới hạn thích nghi thay thế tham số giới hạn tĩnh.
ABC-MNT [72]	Chia quần thể thành nhiều nhóm nhỏ với các cấu trúc liên kết mạng khác nhau, như Von Neumann, Moore hay Hexagonal.
FOABC [75]	Tích hợp thêm phép tính vi tích phân phân số trong phương trình tìm kiếm, sử dụng đặc tính của phép đạo hàm để ghi nhớ kết quả tìm kiếm trong quá khứ, giúp tăng độ chính xác của giải pháp.
FLABC [73]	Lựa chọn phương trình tìm kiếm phù hợp với từng loại địa hình: trơn nhẵn hay gồ ghề; chia quần thể thành nhiều nhóm động cho giai đoạn tìm kiếm; sử dụng toán tử tìm kiếm lân cận toàn cầu cải tiến để bảo lưu kinh nghiệm quá khứ thay vì khởi tạo ngẫu nhiên từ đầu.
DABC [83]	Điều chỉnh phương trình tìm kiếm để xử lý dữ liệu rời rạc của bài toán lập lịch.
ABC-PRM-EP [65]	Khởi tạo quần thể bằng kết quả tìm đường theo phương pháp PRM.
HABC-GA [67]	Tích hợp các toán tử lai ghép và đột biến thích ứng đặc trưng của GA vào giai đoạn tìm kiếm của ABC.
IACO-IABC [66]	Kết hợp sức mạnh tìm kiếm đường đi chi tiết của ACO với khả năng khởi tạo nhanh của ABC, áp dụng cơ chế tối ưu hóa đường đi loại bỏ các nút dư thừa.

Bảng 4.28: Đối chiếu đóng góp khoa học của kỹ thuật DeZigzagPath

Biến thể	Mô tả
ElasticABC	Thiết kế chuyên dụng cho ElasticABC, xử lý vấn đề gấp khúc cục bộ dựa trên cơ chế kiểm tra tham số biên độ "dao động".

Biến thể	Mô tả
ADL-ABC [74]	Sử dụng đa thức bậc 3 để nối các điểm trung gian với nhau.
IABC [78]	Tích hợp chiến lược tuyến tính hóa đường đi trong hàm mục tiêu thông qua kiểm tra tầm nhìn thẳng trong bản đồ lưới, kết hợp trong giai đoạn tìm kiếm.

Hai biến thể có đặc điểm hoạt động gần tương tự như ElasticABC là HABC-GA [67] và IABC [78]. Thuật toán thứ nhất sử dụng toán tử Lai ghép và Đột biến, đặc trưng của GA, tích hợp trong giai đoạn tìm kiếm của ABC; cho phép dịch chuyển điểm trung gian nằm quá gần vùng nguy hiểm sang vị trí an toàn hơn. Thuật toán thứ hai sử dụng hàm mục tiêu tích hợp chiến lược tuyến tính hoá đường đi thông kiểm tra tầm nhìn thẳng trong bản đồ lưới, kết hợp trong giai đoạn tìm kiếm; cho phép loại bỏ các điểm trung gian "thừa" ra khỏi đường đi tìm được, rút ngắn số lượng điểm trung gian cuối cùng. Bảng 4.29 tổng hợp các điểm khác biệt giữa ba biến thể.

Bảng 4.29: Đối chiếu ElasticABC với HABC-GA và IABC

	ElasticABC	HABC-GA	IABC
Lớp bài toán	Hoạch định đường đi cho hệ thống đa robot.	Hoạch định đường đi cho cho một robot di động.	Hoạch định đường đi cho một robot di động.
Yếu tố kỹ thuật cốt lõi	Cặp toán tử đối ngẫu Co và Dẫn hoạt động trong giai đoạn tìm kiếm.	Toán tử Lai ghép và Đột biến của GA.	Chiến lược tuyến tính hóa đường đi với toán tử "cắt".
Hàm mục tiêu	Tổng độ dài Euclid và thành phần phạt va chạm với vật cản.	Tổng độ dài Euclid, độ trơn của hành trình và thành phần phạt va chạm với vật cản.	Tổng độ dài Euclid và thành phần phạt va chạm với vật cản, tích hợp chiến lược tuyến tính hóa đường đi.

	ElasticABC	HABC-GA	IABC
Khởi tạo	Khởi tạo phỏng đoán dựa trên toán tử Dẫn.	Khởi tạo phỏng đoán dựa trên phân nhóm quần thể sử dụng RRT.	Khởi tạo ngẫu nhiên.
Xử lý hậu kỳ	DeZigzagPath được tích hợp xử lý vấn đề gấp khúc cục bộ.	-	-

DCM

Sự kết hợp giữa khả năng "phân cụm mềm" của mô hình GMM với hàm mục toàn phương dựa trên tham số chi phí δ_k đã lựa chọn được cấu hình phân cụm cân bằng về số lượng phân tử trong mỗi cụm. Kết quả thử nghiệm số là minh chứng rõ ràng về tính năng cân bằng tải này của DCM, cho phép mô hình hóa sự phân bố dữ liệu theo xác suất và chấp nhận sự chồng lấn giữa các vùng quản lý theo cụm. Xét về đặc điểm kỹ thuật, DCM tương tự như các phương pháp phân cụm dựa trên thuật toán K-means thuộc phân nhóm thứ ba trong trong phân loại các công trình được khảo sát thuộc mục 1.4.

Bảng 4.30: Đối chiếu DCM với phương pháp sử dụng K-means

	DCM	Phương pháp sử dụng K-means [125] [129] [130] [131] [135]
Lớp bài toán	Phân chia phần tử trong hệ thống đa robot thành các nhóm.	Phân công tác vụ cho hệ thống đa robot.
Đặc điểm	- Số cụm được lựa chọn cố định.	- Số cụm tương ứng với số phần tử robot.

	- Cân bằng số lượng phân tử trong từng cụm.	- Cân bằng khối lượng công việc theo hàm mục tiêu.
Loại dữ liệu đầu vào	Tập hợp điểm giao cắt.	Tập hợp điểm nhiệm vụ.
Hàm mục tiêu	Tối ưu theo tham số chi phí δ_k , chỉ số đánh giá độ phức tạp quản lý của hệ thống.	Tối ưu theo quãng đường di chuyển trong nội bộ cụm hoặc thời gian hoàn thành cụm nhiệm vụ.

Sự phân biệt phân cụm cứng của K-means với phân cụm mềm của GMM trong DCM có nguồn gốc từ lĩnh vực ứng dụng chính của hai giải pháp: xử lý dữ liệu. K-means phân cụm dữ liệu dựa trên khoảng cách, phân chia tập dữ liệu thành những cụm phân biệt sao cho tổng bình phương khoảng cách từ mỗi điểm dữ liệu đến tâm cụm của nó là nhỏ nhất. Ngược lại, GMM phân chia dữ liệu theo xác suất, mỗi điểm dữ liệu không bị gán cứng vào một cụm mà sẽ có xác suất thuộc về từng cụm khác nhau. Bảng 4.30 mô tả sự khác biệt của DCM so với những phương pháp phân cụm khác sử dụng thuật toán K-means, đã được công bố gần đây.

Khả năng mở rộng và triển khai giải pháp

Khi quy mô về số lượng phân tử của hệ thống đa robot tăng lên, tổng thể hoạt động của hệ thống có tác động rất rõ ràng, trong đó mức độ tương tác giữa các phân tử với trạm điều khiển trung tâm được thể hiện rõ ràng nhất. Các chỉ tiêu kỹ thuật chịu ảnh hưởng bao gồm:

- Thời gian xử lý tính toán, liên quan trực tiếp đến độ phức tạp tính toán, xử lý của phần mềm;
- Lưu lượng và băng thông cùng với độ trễ truyền thông, liên quan trực tiếp đến dữ liệu chia sẻ hai chiều giữa các phân tử robot và trạm điều khiển trung tâm (mô hình giao tiếp toàn cục);
- Hiệu suất hoạt động tổng thể gắn liền với những chỉ tiêu như: thời gian hoàn thành nhiệm vụ trung bình, năng suất hệ thống (số lượng nhiệm vụ hoàn thành trên một đơn vị thời gian), tỷ lệ hoàn thành nhiệm vụ, thời gian chờ trung bình v.v...

Bên cạnh đó, mức độ tương tác cục bộ giữa các phần tử robot khi hoạt động trên cùng một không gian chung cũng có tác động nhất định. Tỷ lệ xung đột có xu hướng tăng lên, biểu hiện qua sự gia tăng của tập hợp điểm giao cắt. Do vậy, khả năng mở rộng quy mô gắn liền với việc triển khai giải pháp trong hệ thống đa robot thực tế.

Khi được triển khai, thuật toán ElasticABC và DCM là những giải thuật phần mềm được cài đặt trong trạm điều khiển trung tâm. Mối liên hệ giữa ElasticABC và DCM được minh họa như Hình 3.1, trong đó ElasticABC là giải pháp MRPP cụ thể. Hai giải thuật được phát triển cho hệ thống đa robot với khung thuật toán có khả năng xử lý dữ liệu theo mảng, đáp ứng được việc thay đổi (tăng hoặc giảm quy mô về số lượng). Ngoài ra, DCM được thiết kế với mục đích chuyên biệt cho hệ thống đa robot, chia nhỏ các phần tử robot theo cụm tương quan, cung cấp cơ sở kỹ thuật tin cậy cho việc quản lý ở cấp độ hệ thống. Các thử nghiệm số đã được thực hiện với mô hình 10 phần tử (đối với ElasticABC) và lên đến 50 phần tử (đối với DCM). Khi mở rộng quy mô hệ thống, một số ảnh hưởng có thể xảy ra:

- Ở cấp độ hệ thống, thời gian xử lý tính toán của hai giải thuật dự kiến gia tăng, phù hợp với nhận định về chỉ tiêu kỹ thuật ở trên.

- Ở cấp độ phần tử robot, tác động giới hạn cục bộ trong nội bộ từng cụm do có sự phân chia của DCM dựa trên cơ sở đường đi tìm được từ ElasticABC.

4.7 Kết luận chương 4

Nội dung chương đã tập trung trình bày về thử nghiệm và đánh giá hoạt động của thuật toán ElasticABC và DCM cùng với một số thử nghiệm phụ trợ về mô hình quần thể bày ong tập trung/phi tập trung và đối chứng hoạt động ElasticABC với thuật toán ABC truyền thống. Những kết quả chủ yếu đã đạt được như sau:

- i. ElasticABC với cải tiến cốt lõi là tối ưu cấu trúc kết hợp khởi tạo phỏng đoán đã minh chứng khả năng cân bằng giữa khám phá và khai thác; tìm kiếm được hành trình gần tối ưu và thích nghi hiệu quả với điều kiện môi trường thông qua điều chỉnh có chọn lọc số lượng điểm trung gian. Thuật toán thể hiện tính ổn định cao, có khả

năng xử lý các kịch bản khởi tạo "xấu" hoặc môi trường phức tạp để tìm ra giải pháp khả thi với chi phí thấp nhất.

ii. Kỹ thuật DeZigzagPath được thiết kế chuyên dụng cho ElasticABC, đã minh chứng được khả năng xử lý vấn đề gấp khúc cục bộ trong kết quả tìm kiếm, hoàn thiện được chất lượng hành trình thực tế và thể hiện tính ứng dụng rất cao.

iii. Kết quả đối sánh hoạt động của ElasticABC với thuật toán ABC truyền thống đã khẳng định được chất lượng vượt trội của thuật toán đề xuất: sự cân bằng trong khả năng khám phá và khả năng khai thác; đồng thời giải quyết được triệt để hạn chế của chiến lược tìm kiếm có kích thước cố định.

iv. DCM với cách tiếp cận mềm trong phân cụm đã thể hiện được sự ổn định và tin cậy trong hoạt động, thích nghi linh hoạt với những biến động của môi trường cũng như khi thay đổi quy mô hệ thống. Đồng thời, hình thái cụm có sự giao thoa và chấp nhận chồng lấn, cho phép hấp thụ sự phức tạp của dữ liệu đầu vào và môi trường. Tham số chi phí và hàm mục tiêu toàn phương đã hoạt động như một "công cụ" lựa chọn cấu hình cụm phù hợp. Do đó, DCM là giải pháp phân cụm hoàn toàn phù hợp để làm nền tảng cho việc tổ chức quản lý trong hệ thống đa robot.

Kết quả thử nghiệm đã khẳng định ElasticABC hoàn toàn đáp ứng mục đích nghiên cứu thứ nhất và DCM hoàn toàn đáp ứng mục tiêu nghiên cứu thứ hai của Luận án.

Những công bố liên quan đến Chương 4 bao gồm:

- Về thuật toán ElasticABC: Công trình 4 trong Danh mục các công trình khoa học. Ngoài ra, nghiên cứu sinh đã nộp bản thảo Công trình 7 cho tạp chí Array thuộc nhà xuất bản Elsevier.

- Về phương pháp DCM: Công trình 1 và 3 trong Danh mục các công trình khoa học.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Nghiên cứu sinh đã tiến hành những nghiên cứu khác nhau để phát triển được thuật toán ElasticABC, đảm bảo tối ưu toàn cục trong hoạch định đường đi đồng thời duy trì năng lực tìm kiếm để ứng phó với những biến đổi cục bộ, và phương pháp phân vùng DCM hỗ trợ quản lý hệ thống đa robot, phân chia robot thành những nhóm nhỏ qua đó giảm khối lượng dữ liệu vận hành của hệ thống cần trao đổi. Những đóng góp chính được tóm tắt như sau:

- Tìm hiểu về ứng dụng thực tiễn của thuật toán hoạch định đường đi kết hợp với quy tắc giao thông cụ thể để vận hành robot trong môi trường thực tế. Đồng thời, tập trung nghiên cứu về khía cạnh tối ưu của thuật toán ABC truyền thống, xem xét ở hai mô hình quần thể tập trung và phi tập trung. Kết quả đã được công bố trong Công trình 2 và 4 thuộc Danh mục các công trình khoa học.

- Phát triển biến thể ElasticABC ứng dụng cho hoạch định đường đi hệ thống đa robot, cải tiến thuật toán ABC truyền thống về tối ưu cấu trúc sử dụng cặp toán tử đối ngẫu Co và Dẫn kết hợp chiến lược khởi tạo phỏng đoán, tích hợp kỹ thuật xử lý hậu kỳ chuyên dụng DeZigzagPath. ElasticABC có khả năng cân bằng giữa khám phá và khai thác; đường đi của robot thích nghi với độ phức tạp của môi trường thông qua việc cơ chế tự động điều chỉnh số lượng điểm trung gian. Thuật toán hoạt động ổn định và tin cậy trong điều kiện thử nghiệm mô phỏng với nhiều kịch bản khác nhau, có khả năng xử lý tình huống khởi tạo "xấu" để tìm kiếm giải pháp gần tối ưu với chi phí thấp. Kết quả đã được hoàn thiện trong Công trình 7 thuộc Danh mục các công trình khoa học, đã nộp bản thảo cho tạp chí "Array" thuộc danh mục WoS/Scopus.

- Đề xuất phương pháp phân cụm DCM theo cách tiếp cận phân cụm mềm của mô hình GMM kết hợp thuật toán EM, sử dụng tham số chi phí δ_k và hàm mục tiêu toàn phương f_{obj} để lựa chọn cấu hình cụm có số lượng phần tử tương đối cân bằng. Mô hình phân cụm phân bố theo xác suất và chấp nhận sự chồng lấn linh hoạt. Thông qua các thử nghiệm mô phỏng với nhiều kịch bản khác nhau, DCM đã thể hiện được

tính ổn định, khả năng thích nghi linh hoạt về hình thái cụm và tính độc lập với điều kiện môi trường cũng như quy mô hệ thống. Kết quả đã được công bố trong Công trình 1 và 3 thuộc Danh mục các công trình khoa học.

- Đề xuất hai giải pháp kỹ thuật xác định trạng thái chuyển động cùng với điều chỉnh hướng phát hiện vật cản cho các phần tử trong hệ thống đa robot; giải pháp đã được cấp bằng độc quyền giải pháp hữu ích số 3615 và bằng độc quyền sáng chế số 48666. Chúng thuộc về tính năng nhận thức của phần tử robot và hỗ trợ cho tính năng điều hướng cùng với bài toán hoạch định đường đi. Kết quả đã được công bố trong Công trình 5 và 6 thuộc Danh mục các công trình khoa học.

Hướng phát triển của luận án

Với mỗi phương pháp đề xuất trong luận án, nghiên cứu sinh vẫn đề nghị các vấn đề cần xem xét kỹ lưỡng hơn trong các nghiên cứu tiếp theo. Cụ thể như sau:

- Nghiên cứu cơ chế lựa chọn thích nghi hệ số K, số lượng cụm, trong giải thuật DCM dựa trên các chỉ số định lượng từ dữ liệu đầu vào để tối ưu mô hình.

- Đánh giá hoạt động của giải thuật ElasticABC trong điều kiện môi trường có vật cản biên dạng phức tạp, như đa giác lõm, chữ U hoặc V, xử lý bài toán bất cực trị địa phương.

- Đánh giá hiệu suất hoạt động của giải thuật DCM và ElasticABC khi ứng dụng cho các hệ thống đa robot trong điều kiện vận hành thực tế.

DANH MỤC CÔNG TRÌNH KHOA HỌC

Bài báo tạp chí thuộc danh mục WoS hoặc Scopus

1. **Hung Truong Xuan**, Thang Pham Manh, Nha Nguyen Quang, Hanh Nguyen Thi Hong Hanh, "Dynamic Clustering of Multi-Mobile Robot System using Gaussian Mixture Model", *J. Robot. Control*, vol. 6, no. 5 SE-Articles, pp. 2129–2140, Aug. 2025, doi: 10.18196/jrc.v6i5.27184

Báo cáo hội nghị quốc tế

2. **Hung Truong Xuan**, Pham Manh Thang and Thuc Nguyen Van, "Digitizing Working Space of the Multi Mobile Robots System Using an Approximate Cell Decomposition Method", Proceedings of the 7th International Conference on Engineering Mechanics and Automation (ICEMA7), pp. 3-8, 2023, ISBN: 978-604-357-241-4, doi: 10.15625/vap.2023.0110.
3. **Hung X. Truong**, Thang M. Pham, Nha Q. Nguyen, Hanh T. H. Nguyen and Viet A. Dang, "Evaluation of Spatial Stability and Adaptability in a Dynamic Clustering Method for Multi-Robot Systems", Proceedings of the 8th International Conference on Engineering Mechanics and Automation (ICEMA8), pp. 12-18, 2025, doi: 10.15625/vap.2026.0003. ISBN: 978-604-357-500-2
4. **Hung X. Truong**, Thang M. Pham, Nha Q. Nguyen, Hanh T. H. Nguyen and Viet A. Dang, "Artificial Bee Colony for Multi-Robot Path Planning: A Comparative Study of Population Structures", Proceedings of the 8th International Conference on Engineering Mechanics and Automation (ICEMA8), pp. 19-25, 2025, doi: 10.15625/vap.2026.0004. ISBN: 978-604-357-500-2

Sở hữu trí tuệ

5. Nguyễn Thị Hồng Hạnh (VN); **Trương Xuân Hùng** (VN); Nguyễn Đức Minh (VN); Bằng độc quyền sáng chế số 48666 "Cụm cơ cấu điều chỉnh hướng và robot vận chuyển hàng hoá trong nhà kho bao gồm cụm cơ cấu điều chỉnh hướng này", 2025.
6. **Trương Xuân Hùng** (VN); Nguyễn Thị Hồng Hạnh (VN); Trịnh Thăng Long (VN); Nguyễn Văn Thức (VN); Bằng độc quyền giải pháp hữu ích số 3615 "Phương pháp xác định trạng thái chuyển động của robot di động vận chuyển hàng hoá để phục vụ điều khiển và giám sát robot", 2024.

Bài báo tạp chí nộp cho tạp chí "Array" thuộc danh mục WoS/Scopus

7. An Elastic Artificial Bee Colony Algorithm with Dynamic Dimensionality for Multi-Robot Path Planning

DANH MỤC KẾT QUẢ NGHIÊN CỨU KHÁC

Bài báo tạp chí thuộc danh mục WoS hoặc Scopus

Thang M. Pham, Nha Q. Nguyen, Manh V. Hoang, Viet A. Dang, Nam Do, **Hung X. Truong**, Nam D. Vu, “High-Throughput UAV Video Processing: A Multithreaded Architecture for Real-Time Deep Learning-Based Image Analysis,” J. Robot. Control, vol. 6, no. 6, pp. 2746–2759, 2025, doi: 10.18196/jrc.v6i6.28366.

Thang M. Pham, Manh V. Hoang, Ha T. T. Pham, Ho N. Pham, Lam T. Bui, Nam Do, **Hung X. Truong**, Truong Q. Nguyen, “ResPhaseNet: A Residual U-Net Approach for Seismic Phase Detection,” J. Robot. Control, vol. 6, no. 6, pp. 2602–2612, 2025, doi: 10.18196/jrc.v6i6.28384.

Thang M. Pham, Manh V. Hoang, Ha T. T. Pham, Ho N. Pham, Lam T. Bui, Nam Do, **Hung X. Truong**, Truong Q. Ho, “A CNN–LSTM Framework for Time-Windowed Detection of Local Earthquakes from Ocean Bottom Seismometer Data,” J. Robot. Control, vol. 6, no. 6 SE-Articles, pp. 2814–2825, Dec. 2025, doi: 10.18196/jrc.v6i6.28381.

Bài báo tạp chí trong nước

Nguyễn Chính Kiên, Nguyễn Thị Hồng Hạnh, và **Trương Xuân Hùng**, “Thử nghiệm thuật toán tìm đường đi tối ưu cho hệ thống robot di động vận chuyển hàng trong kho tự động,” Bản B của Tạp chí Khoa học và Công nghệ Việt Nam, vol. 67, no. 11 SE-, Nov. 2025, doi: 10.31276/VJST.67(11).39-46.

Bài báo hội nghị trong nước

Nguyen Thi Hong Hanh, Nguyen Duc Minh, **Truong Xuan Hung**, Le Chi Cong, "Development on the lifting mechanism for mobile robot on transportation in the warehouse", Tuyển tập công trình khoa học hội nghị cơ

học toàn quốc kỷ niệm 45 năm thành lập viện cơ học, tr. 686-693, 2024, ISBN: 978-604-357-277-3.

Trịnh Văn Khang, **Trương Xuân Hùng**, Nguyễn Đức Minh, Bùi Nam Dương, Nguyễn Văn Thúc, Trịnh Thăng Long, Nguyễn Thị Hồng Hạnh, "Phân tích và đánh giá về một phương án thiết kế cho khung cấu trúc robot hình cầu bằng phương pháp phần tử hữu hạn", Kỷ yếu Hội nghị Khoa học kỷ niệm 50 năm thành lập Viện Hàn lâm KH&CN Việt Nam, Tiểu ban Công nghệ thông tin - Tự động hoá - Công nghệ vũ trụ, tr. 299-306, 2025, ISBN: 978-604-357-359-6.

TÀI LIỆU THAM KHẢO

- [1] O. Khatib, “Real-time obstacle avoidance for manipulators and mobile robots,” in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, Mar. 1985, vol. 2, pp. 500–505, doi: 10.1109/ROBOT.1985.1087247.
- [2] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*. New York: Oxford University Press, 1999.
- [3] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of ICNN’95 - International Conference on Neural Networks*, Nov. 1995, vol. 4, pp. 1942–1948 vol.4, doi: 10.1109/ICNN.1995.488968.
- [4] A. Colomi, M. Dorigo, and V. Maniezzo, “Distributed Optimization by ant colonies,” *Proc. First Eur. Conf. Artif. Life*, no. or D, pp. 134–142, 1991.
- [5] D. Karaboga, “An Idea Based on Honey Bee Swarm for Numerical Optimization, Technical Report - TR06,” *Tech. Report, Erciyes Univ.*, 2005.
- [6] A. R. Mehrabian and C. Lucas, “A novel numerical optimization algorithm inspired from weed colonization,” *Ecol. Inform.*, vol. 1, no. 4, pp. 355–366, 2006, doi: 10.1016/j.ecoinf.2006.07.003.
- [7] S. Saremi, S. Mirjalili, and A. Lewis, “Grasshopper Optimisation Algorithm: Theory and application,” *Adv. Eng. Softw.*, vol. 105, pp. 30–47, 2017, doi: 10.1016/j.advengsoft.2017.01.004.
- [8] F. Rubio, F. Valero, and C. Llopis-Albert, “A review of mobile robots: Concepts, methods, theoretical framework, and applications,” *Int. J. Adv. Robot. Syst.*, vol. 16, no. 2, p. 1729881419839596, 2019, doi: 10.1177/1729881419839596.
- [9] R. N. Darmanin and M. K. Bugeja, “A review on multi-robot systems categorised by application domain,” in *2017 25th Mediterranean Conference on Control and Automation (MED)*, 2017, pp. 701–706, doi:

10.1109/MED.2017.7984200.

- [10] Y. Cai and S. X. Yang, “A Survey on multi-robot systems,” in *World Automation Congress 2012*, 2012, pp. 1–6.
- [11] P. G. F. Dias, M. C. Silva, G. P. Rocha Filho, P. A. Vargas, L. P. Cota, and G. Pessin, “Swarm Robotics: A Perspective on the Latest Reviewed Concepts and Applications,” *Sensors*, vol. 21, no. 6, 2021, doi: 10.3390/s21062062.
- [12] E. Osaba, J. Del Ser, A. Iglesias, and X.-S. Yang, “Soft Computing for Swarm Robotics: New Trends and Applications,” *J. Comput. Sci.*, vol. 39, p. 101049, 2020, doi: 10.1016/j.jocs.2019.101049.
- [13] D. Albiero, A. Pontin Garcia, C. Kiyoshi Umezu, and R. Leme de Paulo, “Swarm robots in mechanized agricultural operations: A review about challenges for research,” *Comput. Electron. Agric.*, vol. 193, p. 106608, 2022, doi: 10.1016/j.compag.2021.106608.
- [14] C. Ju, J. Kim, J. Seol, and H. Il Son, “A review on multirobot systems in agriculture,” *Comput. Electron. Agric.*, vol. 202, p. 107336, 2022, doi: 10.1016/j.compag.2022.107336.
- [15] D. S. Drew, “Multi-Agent Systems for Search and Rescue Applications,” *Curr. Robot. Reports*, vol. 2, no. 2, pp. 189–200, 2021, doi: 10.1007/s43154-021-00048-3.
- [16] J. P. Queralta *et al.*, “Collaborative Multi-Robot Search and Rescue: Planning, Coordination, Perception, and Active Vision,” *IEEE Access*, vol. 8, pp. 191617–191643, 2020, doi: 10.1109/ACCESS.2020.3030190.
- [17] Z. Zhou, J. Liu, and J. Yu, “A Survey of Underwater Multi-Robot Systems,” *IEEE/CAA J. Autom. Sin.*, vol. 9, no. 1, pp. 1–18, 2022, doi: 10.1109/JAS.2021.1004269.
- [18] H. Kabir, M.-L. Tham, and Y. C. Chang, “Internet of robotic things for mobile robots: Concepts, technologies, challenges, applications, and future directions,” *Digit. Commun. Networks*, vol. 9, no. 6, pp. 1265–1290, 2023, doi:

10.1016/j.dcan.2023.05.006.

- [19] N. AbuJabal, T. Rabie, M. Baziyad, I. Kamel, and K. Almazrouei, "Path Planning Techniques for Real-Time Multi-Robot Systems: A Systematic Review," *Electronics*, vol. 13, no. 12, 2024, doi: 10.3390/electronics13122239.
- [20] S. Banik, S. C. Banik, and S. S. Mahmud, "Path Planning Approaches in Multi-robot System: A Review," *Eng. Reports*, vol. 7, no. 1, p. e13035, Jan. 2025, doi: 10.1002/eng2.13035.
- [21] S. Lin, A. Liu, J. Wang, and X. Kong, "A Review of Path-Planning Approaches for Multiple Mobile Robots," *Machines*, vol. 10, no. 9, 2022, doi: 10.3390/machines10090773.
- [22] H. Ma, "Graph-Based Multi-Robot Path Finding and Planning," *Curr. Robot. Reports*, vol. 3, no. 3, pp. 77–84, 2022, doi: 10.1007/s43154-022-00083-8.
- [23] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, and P. Wang, "Path planning techniques for mobile robots: Review and prospect," *Expert Syst. Appl.*, vol. 227, p. 120254, 2023, doi: 10.1016/j.eswa.2023.120254.
- [24] J. A. Abdulsahab and D. J. Kadhim, "Classical and Heuristic Approaches for Mobile Robot Path Planning: A Survey," *Robotics*, vol. 12, no. 4, 2023, doi: 10.3390/robotics12040093.
- [25] A. Loganathan and N. S. Ahmad, "A systematic review on recent advances in autonomous mobile robot navigation," *Eng. Sci. Technol. an Int. J.*, vol. 40, p. 101343, 2023, doi: 10.1016/j.jestch.2023.101343.
- [26] M. N. A. Wahab, S. Nefti-Meziani, and A. Atyabi, "A comparative review on mobile robot path planning: Classical or meta-heuristic methods?," *Annu. Rev. Control*, vol. 50, pp. 233–252, 2020, doi: 10.1016/j.arcontrol.2020.10.001.
- [27] L. Dong, Z. He, C. Song, and C. Sun, "A review of mobile robot motion planning methods: from classical motion planning workflows to reinforcement learning-based architectures," *J. Syst. Eng. Electron.*, vol. 34, no. 2, pp. 439–459, 2023, doi: 10.23919/JSEE.2023.000051.

- [28] H. Sun, W. Zhang, R. Yu, and Y. Zhang, “Motion Planning for Mobile Robots—Focusing on Deep Reinforcement Learning: A Systematic Review,” *IEEE Access*, vol. 9, pp. 69061–69081, 2021, doi: 10.1109/ACCESS.2021.3076530.
- [29] C. Zhou, B. Huang, and P. Fränti, “A review of motion planning algorithms for intelligent robots,” *J. Intell. Manuf.*, vol. 33, no. 2, pp. 387–424, 2022, doi: 10.1007/s10845-021-01867-z.
- [30] A. Orthey, C. Chamzas, and L. E. Kavraki, “Sampling-Based Motion Planning: A Comparative Review,” *Annu. Rev. Control. Robot. Auton. Syst.*, vol. 7, no. Volume 7, 2024, pp. 285–310, 2024, doi: 10.1146/annurev-control-061623-094742.
- [31] R. Almadhoun, T. Taha, L. Seneviratne, and Y. Zweiri, “A survey on multi-robot coverage path planning for model reconstruction and mapping,” *SN Appl. Sci.*, vol. 1, no. 8, p. 847, 2019, doi: 10.1007/s42452-019-0872-y.
- [32] C. S. Tan, R. Mohd-Mokhtar, and M. R. Arshad, “A Comprehensive Review of Coverage Path Planning in Robotics Using Classical and Heuristic Algorithms,” *IEEE Access*, vol. 9, pp. 119310–119342, 2021, doi: 10.1109/ACCESS.2021.3108177.
- [33] M. Zhou, J. Li, C. Wang, J. Wang, and L. Wang, “Applications of Voronoi Diagrams in Multi-Robot Coverage: A Review,” *J. Mar. Sci. Eng.*, vol. 12, no. 6, 2024, doi: 10.3390/jmse12061022.
- [34] S. Cohen and N. Agmon, “Recent Advances in Formations of Multiple Robots,” *Curr. Robot. Reports*, vol. 2, no. 2, pp. 159–175, 2021, doi: 10.1007/s43154-021-00049-2.
- [35] A. S. Aguiar, F. N. dos Santos, J. B. Cunha, H. Sobreira, and A. J. Sousa, “Localization and Mapping for Robots in Agriculture and Forestry: A Survey,” *Robotics*, vol. 9, no. 4. 2020, doi: 10.3390/robotics9040097.
- [36] Y. Liu, S. Wang, Y. Xie, T. Xiong, and M. Wu, “A Review of Sensing

- Technologies for Indoor Autonomous Mobile Robots,” *Sensors*, vol. 24, no. 4, 2024, doi: 10.3390/s24041222.
- [37] M. A. K. Niloy *et al.*, “Critical Design and Control Issues of Indoor Autonomous Mobile Robots: A Review,” *IEEE Access*, vol. 9, pp. 35338–35370, 2021, doi: 10.1109/ACCESS.2021.3062557.
 - [38] R. B. Sousa, H. M. Sobreira, and A. P. Moreira, “A systematic literature review on long-term localization and mapping for mobile robots,” *J. F. Robot.*, vol. 40, no. 5, pp. 1245–1322, Aug. 2023, doi: 10.1002/rob.22170.
 - [39] J. Gielis, A. Shankar, and A. Prorok, “A Critical Review of Communications in Multi-robot Systems,” *Curr. Robot. Reports*, vol. 3, no. 4, pp. 213–225, 2022, doi: 10.1007/s43154-022-00090-9.
 - [40] L. C. Garaffa, M. Basso, A. A. Konzen, and E. P. de Freitas, “Reinforcement Learning for Mobile Robotics Exploration: A Survey,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 34, no. 8, pp. 3796–3810, 2023, doi: 10.1109/TNNLS.2021.3124466.
 - [41] J. Orr and A. Dutta, “Multi-Agent Deep Reinforcement Learning for Multi-Robot Applications: A Survey,” *Sensors*, vol. 23, no. 7, 2023, doi: 10.3390/s23073625.
 - [42] K. Zhu and T. Zhang, “Deep reinforcement learning based mobile robot navigation: A review,” *Tsinghua Sci. Technol.*, vol. 26, no. 5, pp. 674–691, 2021, doi: 10.26599/TST.2021.9010012.
 - [43] P. E. Hart, N. J. Nilsson, and B. Raphael, “A Formal Basis for the Heuristic Determination of Minimum Cost Paths,” *IEEE Trans. Syst. Sci. Cybern.*, vol. 4, no. 2, pp. 100–107, 1968, doi: 10.1109/TSSC.1968.300136.
 - [44] J.-C. Latombe, “Exact Cell Decomposition and Approximate Cell Decomposition,” in *Robot Motion Planning*, Springer New York, 2012, pp. 200–294.
 - [45] O. A. A. Salama, M. E. H. Eltaib, H. A. Mohamed, and O. Salah, “RCD: Radial

- Cell Decomposition Algorithm for Mobile Robot Path Planning,” *IEEE Access*, vol. 9, pp. 149982–149992, 2021, doi: 10.1109/ACCESS.2021.3125105.
- [46] S. K. Debnath *et al.*, “Different Cell Decomposition Path Planning Methods for Unmanned Air Vehicles-A Review,” in *Proceedings of the 11th National Technical Seminar on Unmanned System Technology 2019*, 2021, pp. 99–111, doi: 10.1007/978-981-15-5281-6_8.
- [47] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, “Probabilistic roadmaps for path planning in high-dimensional configuration spaces,” *IEEE Trans. Robot. Autom.*, vol. 12, no. 4, pp. 566–580, 1996, doi: 10.1109/70.508439.
- [48] S. M. H. Rostami, A. K. Sangaiah, J. Wang, and X. Liu, “Obstacle avoidance of mobile robots using modified artificial potential field algorithm,” *EURASIP J. Wirel. Commun. Netw.*, vol. 2019, no. 1, p. 70, 2019, doi: 10.1186/s13638-019-1396-2.
- [49] Z. Wu, W. Su, and J. Li, “Multi-robot path planning based on improved artificial potential field and B-spline curve optimization,” in *2019 Chinese Control Conference (CCC)*, Jul. 2019, pp. 4691–4696, doi: 10.23919/ChiCC.2019.8865232.
- [50] F. Matoui, B. Boussaid, B. Metoui, and M. N. Abdelkrim, “Contribution to the path planning of a multi-robot system: centralized architecture,” *Intell. Serv. Robot.*, vol. 13, no. 1, pp. 147–158, 2020, doi: 10.1007/s11370-019-00302-w.
- [51] T. Zhao, H. Li, and S. Dian, “Multi-robot path planning based on improved artificial potential field and fuzzy inference system,” *J. Intell. Fuzzy Syst.*, vol. 39, pp. 7621–7637, 2020, doi: 10.3233/JIFS-200869.
- [52] Z. Pan, D. Li, K. Yang, and H. Deng, “Multi-Robot Obstacle Avoidance Based on the Improved Artificial Potential Field and PID Adaptive Tracking Control Algorithm,” *Robotica*, vol. 37, no. 11, pp. 1883–1903, 2019, doi: 10.1017/S026357471900033X.

- [53] K. F. E. Tsang, Y. Ni, C. F. R. Wong, and L. Shi, “A Novel Warehouse Multi-Robot Automation System with Semi-Complete and Computationally Efficient Path Planning and Adaptive Genetic Task Allocation Algorithms,” in *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, Nov. 2018, pp. 1671–1676, doi: 10.1109/ICARCV.2018.8581092.
- [54] F. Matoui, B. Boussaid, and M. N. Abdelkrim, “Distributed path planning of a multi-robot system based on the neighborhood artificial potential field approach,” *Simulation*, vol. 95, no. 7, pp. 637–657, Jul. 2018, doi: 10.1177/0037549718785440.
- [55] F. Metoui, B. Boussaid, and M. N. Abdelkrim, “Path Planning for a Multi-robot System with Decentralized Control Architecture BT - New Trends in Robot Control,” J. Ghommam, N. Derbel, and Q. Zhu, Eds. Singapore: Springer Singapore, 2020, pp. 229–259.
- [56] S. S. Ge and Y. J. Cui, “Dynamic Motion Planning for Mobile Robots Using Potential Field Method,” *Auton. Robots*, vol. 13, no. 3, pp. 207–222, 2002, doi: 10.1023/A:1020564024509.
- [57] L. Yin, Y. Yin, and C. J. Lin, “A new potential field method for mobile robot path planning in the dynamic environments,” *Asian J. Control*, vol. 11, no. 2, pp. 214–225, 2009, doi: 10.1002/asjc.98.
- [58] S. Huang, R. S. H. Teo, and K. K. Tan, “Collision avoidance of multi unmanned aerial vehicles: A review,” *Annu. Rev. Control*, vol. 48, pp. 147–164, 2019, doi: 10.1016/j.arcontrol.2019.10.001.
- [59] A. N. A. Rafai, N. Adzhar, and N. I. Jaini, “A Review on Path Planning and Obstacle Avoidance Algorithms for Autonomous Mobile Robots,” *J. Robot.*, vol. 2022, no. 1, p. 2538220, 2022, doi: 10.1155/2022/2538220.
- [60] L. V. Nguyen, “Swarm Intelligence-Based Multi-Robotics: A Comprehensive Review,” *AppliedMath*, vol. 4, no. 4, pp. 1192–1210, 2024, doi: 10.3390/appliedmath4040064.

- [61] D. Karaboga and B. Basturk, “A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm,” *J. Glob. Optim.*, vol. 39, no. 3, pp. 459–471, 2007, doi: 10.1007/s10898-007-9149-x.
- [62] D. Karaboga and B. Akay, “A comparative study of Artificial Bee Colony algorithm,” *Appl. Math. Comput.*, vol. 214, no. 1, pp. 108–132, 2009, doi: 10.1016/j.amc.2009.03.090.
- [63] A. Q. Faridi, S. Sharma, A. Shukla, R. Tiwari, and J. Dhar, “Multi-robot multi-target dynamic path planning using artificial bee colony and evolutionary programming in unknown environment,” *Intell. Serv. Robot.*, vol. 11, no. 2, pp. 171–186, 2018, doi: 10.1007/s11370-017-0244-7.
- [64] S. Kumar and A. Sikander, “Optimum Mobile Robot Path Planning Using Improved Artificial Bee Colony Algorithm and Evolutionary Programming,” *Arab. J. Sci. Eng.*, vol. 47, no. 3, pp. 3519–3539, 2022, doi: 10.1007/s13369-021-06326-8.
- [65] S. Kumar and A. Sikander, “A novel hybrid framework for single and multi-robot path planning in a complex industrial environment,” *J. Intell. Manuf.*, vol. 35, no. 2, pp. 587–612, 2024, doi: 10.1007/s10845-022-02056-2.
- [66] G. Li, C. Liu, L. Wu, and W. Xiao, “A mixing algorithm of ACO and ABC for solving path planning of mobile robot,” *Appl. Soft Comput.*, vol. 148, p. 110868, 2023, doi: 10.1016/j.asoc.2023.110868.
- [67] F. Ye, P. Duan, L. Meng, and L. Xue, “A hybrid artificial bee colony algorithm with genetic augmented exploration mechanism toward safe and smooth path planning for mobile robot,” *Biomim. Intell. Robot.*, vol. 5, no. 2, p. 100206, 2025, doi: 10.1016/j.birob.2024.100206.
- [68] S. Wang, Y. Zhu, Y. Du, and M. Yang, “Research on Path Planning Method for Mobile Platforms Based on Hybrid Swarm Intelligence Algorithms in Multi-Dimensional Environments,” *Biomimetics*, vol. 10, no. 8, 2025, doi: 10.3390/biomimetics10080503.

- [69] Y. Cui, W. Hu, and A. Rahmani, “Multi-robot path planning using learning-based Artificial Bee Colony algorithm,” *Eng. Appl. Artif. Intell.*, vol. 129, p. 107579, 2024, doi: 10.1016/j.engappai.2023.107579.
- [70] F. Ye, P. Duan, L. Meng, H. Sang, and K. Gao, “An enhanced artificial bee colony algorithm with self-learning optimization mechanism for multi-objective path planning problem,” *Eng. Appl. Artif. Intell.*, vol. 149, p. 110444, 2025, doi: 10.1016/j.engappai.2025.110444.
- [71] F. Xu *et al.*, “A new global best guided artificial bee colony algorithm with application in robot path planning,” *Appl. Soft Comput.*, vol. 88, p. 106037, 2020, doi: 10.1016/j.asoc.2019.106037.
- [72] X. Zhou, Y. Wu, M. Zhong, and M. Wang, “Artificial bee colony algorithm based on multiple neighborhood topologies,” *Appl. Soft Comput.*, vol. 111, p. 107697, 2021, doi: 10.1016/j.asoc.2021.107697.
- [73] X. Zhou, J. Song, S. Wu, and M. Wang, “Artificial bee colony algorithm based on online fitness landscape analysis,” *Inf. Sci. (Ny)*, vol. 619, pp. 603–629, 2023, doi: 10.1016/j.ins.2022.11.056.
- [74] R. Kamil, M. Mohamed, and B. Oleiwi, “Path Planning of Mobile Robot Using Improved Artificial Bee Colony Algorithm,” *Eng. Technol. J.*, vol. 38, no. 9, pp. 1384–1395, 2020, doi: 10.30684/etj.v38i9a.1100.
- [75] Y. Cui, W. Hu, and A. Rahmani, “Fractional-order artificial bee colony algorithm with application in robot path planning,” *Eur. J. Oper. Res.*, vol. 306, no. 1, pp. 47–64, 2023, doi: 10.1016/j.ejor.2022.11.007.
- [76] X. Chai *et al.*, “UAV Path Planners in Complex Environments: A Multi-Dimensional Perturbation Based on Artificial Bee Colony,” *IEEE Access*, vol. 13, pp. 110090–110109, 2025, doi: 10.1109/ACCESS.2025.3582770.
- [77] Y. Liang, F. Meng, L. Sun, Y. Zhang, J. He, and K. Yan, “An Improved Artificial Bee Colony Algorithm for Dynamic UAV Trajectory Planning,” in *2025 8th International Conference on Robotics, Control and Automation*

- Engineering (RCAE)*, 2025, pp. 241–246, doi: 10.1109/RCAE66389.2025.11355240.
- [78] M. Y. Yildirim and R. Akay, “An efficient grid-based path planning approach using improved artificial bee colony algorithm,” *Knowledge-Based Syst.*, vol. 318, p. 113528, 2025, doi: 10.1016/j.knosys.2025.113528.
- [79] N. Promkaew, S. Thammawiset, P. Srisan, P. Sanitchon, T. Tummawai, and S. Sukpancharoen, “Development of metaheuristic algorithms for efficient path planning of autonomous mobile robots in indoor environments,” *Results Eng.*, vol. 22, p. 102280, 2024, doi: 10.1016/j.rineng.2024.102280.
- [80] J.-Y. Chen, Q.-K. Pan, J. S. Neufeld, and Z.-H. Miao, “Optimization of task assignment for multi-farm multi-weeding robots based on discrete artificial bee colony algorithm,” *Expert Syst. Appl.*, vol. 267, p. 126182, 2025, doi: 10.1016/j.eswa.2024.126182.
- [81] L.-L. Dai, Q.-K. Pan, Z.-H. Miao, P. N. Suganthan, and K.-Z. Gao, “Multi-Objective Multi-Picking-Robot Task Allocation: Mathematical Model and Discrete Artificial Bee Colony Algorithm,” *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 6, pp. 6061–6073, 2024, doi: 10.1109/TITS.2023.3336659.
- [82] H. Guo, Z. Miao, J. C. Ji, and Q. Pan, “An effective collaboration evolutionary algorithm for multi-robot task allocation and scheduling in a smart farm,” *Knowledge-Based Syst.*, vol. 289, p. 111474, 2024, doi: 10.1016/j.knosys.2024.111474.
- [83] Y. Pan and Z. Miao, “Smart Farm Picking and Distribution Integration : Discrete Artificial Bee Colony Algorithm,” in *2024 5th International Conference on Information Science, Parallel and Distributed Systems (ISPDS)*, 2024, pp. 535–539, doi: 10.1109/ISPDS62779.2024.10667521.
- [84] P. Zhang, M. Wang, G. Zhang, P. Zheng, M. Jin, and J. Zhang, “Multi-robot multi-station welding flow shop closed-loop rescheduling with deep reinforcement learning and improved artificial bee colony algorithm,” *Comput.*

- Ind. Eng.*, vol. 193, p. 110295, 2024, doi: 10.1016/j.cie.2024.110295.
- [85] J. Guo, Y. Li, B. Du, X. Sun, and K. Wang, “A multi-population cooperative coevolution artificial bee colony algorithm for partial multi-robotic disassembly line balancing problem considering preventive maintenance scenarios,” *Adv. Eng. Informatics*, vol. 62, p. 102750, 2024, doi: 10.1016/j.aei.2024.102750.
 - [86] Y. Ren, K. Gao, Y. Fu, D. Li, and P. N. Suganthan, “Ensemble artificial bee colony algorithm with Q-learning for scheduling Bi-objective disassembly line,” *Appl. Soft Comput.*, vol. 155, p. 111415, 2024, doi: 10.1016/j.asoc.2024.111415.
 - [87] C. Xu, H. Wei, X. Guo, S. Liu, L. Qi, and Z. Zhao, “Human-Robot Collaboration Multi-objective Disassembly Line Balancing Subject to Task Failure via Multi-objective Artificial Bee Colony Algorithm,” *IFAC-PapersOnLine*, vol. 53, no. 5, pp. 1–6, 2020, doi: 10.1016/j.ifacol.2021.04.076.
 - [88] R. Szczepanski, K. Erwinski, M. Tejer, A. Bereit, and T. Tarczewski, “Optimal scheduling for palletizing task using robotic arm and artificial bee colony algorithm,” *Eng. Appl. Artif. Intell.*, vol. 113, p. 104976, 2022, doi: 10.1016/j.engappai.2022.104976.
 - [89] F. Zitouni, R. Maamri, and S. Harous, “FA–QABC–MRTA: a solution for solving the multi-robot task allocation problem,” *Intell. Serv. Robot.*, vol. 12, no. 4, pp. 407–418, 2019, doi: 10.1007/s11370-019-00291-w.
 - [90] Z. Zhuang, Z. Huang, Y. Sun, and W. Qin, “Optimization for cooperative task planning of heterogeneous multi-robot systems in an order picking warehouse,” *Eng. Optim.*, vol. 53, no. 10, pp. 1715–1732, Oct. 2021, doi: 10.1080/0305215X.2020.1821198.
 - [91] H.-C. Huang and C.-C. Chuang, “Artificial Bee Colony Optimization Algorithm Incorporated With Fuzzy Theory for Real-Time Machine Learning Control of Articulated Robotic Manipulators,” *IEEE Access*, vol. 8, pp.

- 192481–192492, 2020, doi: 10.1109/ACCESS.2020.3032715.
- [92] Y. M. Ayid, M. Zakaraia, and M. M. Eltoukhy, “An artificial bee colony optimization algorithms for solving fuzzy capacitated logistic distribution center problem,” *MethodsX*, vol. 13, p. 102964, 2024, doi: 10.1016/j.mex.2024.102964.
 - [93] H. Hakli, “The optimization of wind turbine placement using a binary artificial bee colony algorithm with multi-dimensional updates,” *Electr. Power Syst. Res.*, vol. 216, p. 109094, 2023, doi: 10.1016/j.epsr.2022.109094.
 - [94] H. Su *et al.*, “Horizontal and vertical search artificial bee colony for image segmentation of COVID-19 X-ray images,” *Comput. Biol. Med.*, vol. 142, p. 105181, 2022, doi: 10.1016/j.compbiomed.2021.105181.
 - [95] A. Abbadi and R. Matousek, “Hybrid rule-based motion planner for mobile robot in cluttered workspace,” *Soft Comput.*, vol. 22, no. 6, pp. 1815–1831, 2018, doi: 10.1007/s00500-016-2103-4.
 - [96] R. Bähnemann, N. Lawrance, J. J. Chung, M. Pantic, R. Siegwart, and J. Nieto, “Revisiting Boustrophedon Coverage Path Planning as a Generalized Traveling Salesman Problem BT - Field and Service Robotics,” in *Proceedings in Advanced Robotics*, 2021, pp. 277–290.
 - [97] A. Abbadi and V. Prenosil, “Safe Path Planning Using Cell Decomposition,” in *International Conference Distance Learning, Simulation and Communication*, 2015, no. May, pp. 8–14.
 - [98] T. Arney, “An efficient solution to autonomous path planning by Approximate Cell Decomposition,” in *2007 Third International Conference on Information and Automation for Sustainability*, 2007, pp. 88–93, doi: 10.1109/ICIAFS.2007.4544785.
 - [99] K. R. Guruprasad and T. D. Ranjitha, “CPC Algorithm: Exact Area Coverage by a Mobile Robot Using Approximate Cellular Decomposition,” *Robotica*, vol. 39, no. 7, pp. 1141–1162, 2021, doi: 10.1017/S026357472000096X.

- [100] V. G. Nair and K. R. Guruprasad, "MR-SimExCoverage: Multi-robot Simultaneous Exploration and Coverage," *Comput. Electr. Eng.*, vol. 85, p. 106680, 2020, doi: 10.1016/j.compeleceng.2020.106680.
- [101] X. Huang, M. Sun, H. Zhou, and S. Liu, "A Multi-Robot Coverage Path Planning Algorithm for the Environment With Multiple Land Cover Types," *IEEE Access*, vol. 8, pp. 198101–198117, 2020, doi: 10.1109/ACCESS.2020.3027422.
- [102] S. W. Cho, H. J. Park, H. Lee, D. H. Shim, and S.-Y. Kim, "Coverage path planning for multiple unmanned aerial vehicles in maritime search and rescue operations," *Comput. Ind. Eng.*, vol. 161, p. 107612, 2021, doi: 10.1016/j.cie.2021.107612.
- [103] I. Iswanto, O. Wahyunggoro, and A. I. Cahyadi, "Quadrotor Path Planning Based On Modified Fuzzy Cell Decomposition Algorithm," *TELKOMNIKA*, vol. 14, no. 2, pp. 655–664, 2016, doi: 10.12928/TELKOMNIKA.v14i1.2989.
- [104] Z. E. Kanoon, A. S. Al-Araji, and M. N. Abdullah, "Enhancement of Cell Decomposition Path-Planning Algorithm for Autonomous Mobile Robot Based on an Intelligent Hybrid Optimization Method," *Int. J. Intell. Eng. Syst.*, vol. 15, no. 3, pp. 161–175, 2022, doi: 10.22266/ijies2022.0630.14.
- [105] P. T. Kyaw, A. Paing, T. T. Thu, R. E. Mohan, A. Vu Le, and P. Veerajagadheswar, "Coverage Path Planning for Decomposition Reconfigurable Grid-Maps Using Deep Reinforcement Learning Based Travelling Salesman Problem," *IEEE Access*, vol. 8, pp. 225945–225956, 2020, doi: 10.1109/ACCESS.2020.3045027.
- [106] F. Lingelbach, "Path planning using probabilistic cell decomposition," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, 2004, vol. 1, pp. 467–472 Vol.1, doi: 10.1109/ROBOT.2004.1307193.
- [107] R. V Cowlagi and P. Tsiotras, "Beyond quadtrees: Cell decompositions for path

- planning using wavelet transforms,” in *2007 46th IEEE Conference on Decision and Control*, 2007, pp. 1392–1397, doi: 10.1109/CDC.2007.4434146.
- [108] C. Cai and S. Ferrari, “Information-Driven Sensor Path Planning by Approximate Cell Decomposition,” *IEEE Trans. Syst. Man, Cybern. Part B*, vol. 39, no. 3, pp. 672–689, 2009, doi: 10.1109/TSMCB.2008.2008561.
- [109] R. J. Alitappeh and K. Jeddisaravi, “Multi-robot exploration in task allocation problem,” *Appl. Intell.*, vol. 52, no. 2, pp. 2189–2211, 2022, doi: 10.1007/s10489-021-02483-3.
- [110] V. G. Nair and K. R. Guruprasad, “2D-VPC: An Efficient Coverage Algorithm for Multiple Autonomous Vehicles,” *Int. J. Control. Autom. Syst.*, vol. 19, no. 8, pp. 2891–2901, 2021, doi: 10.1007/s12555-020-0389-6.
- [111] J. Kim and H. Il Son, “A Voronoi Diagram-Based Workspace Partition for Weak Cooperation of Multi-Robot System in Orchard,” *IEEE Access*, vol. 8, pp. 20676–20686, 2020, doi: 10.1109/ACCESS.2020.2969449.
- [112] V. G. Nair, R. S. Adarsh, K. P. Jayalakshmi, M. V Dileep, and K. R. Guruprasad, “Cooperative Online Workspace Allocation in the Presence of Obstacles for Multi-robot Simultaneous Exploration and Coverage Path Planning Problem,” *Int. J. Control. Autom. Syst.*, vol. 21, no. 7, pp. 2338–2349, 2023, doi: 10.1007/s12555-022-0182-9.
- [113] Ayan Dutta, Amitabh Bhattacharya, O Patrick Kreidl, Anirban Ghosh, and Prithviraj Dasgupta, “Multi-robot informative path planning in unknown environments through continuous region partitioning,” *Int. J. Adv. Robot. Syst.*, vol. 17, no. 6, p. 1729881420970461, Nov. 2020, doi: 10.1177/1729881420970461.
- [114] P. Bhattacharya and M. L. Gavrilova, “Density-Based Clustering Based on Topological Properties of the Data Set BT - Generalized Voronoi Diagram: A Geometry-Based Approach to Computational Intelligence,” M. L. Gavrilova, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 197–214.

- [115] J. G. Martin, F. J. Muros, J. M. Maestre, and E. F. Camacho, “Multi-robot task allocation clustering based on game theory,” *Rob. Auton. Syst.*, vol. 161, p. 104314, 2023, doi: 10.1016/j.robot.2022.104314.
- [116] J. Hu, P. Bhowmick, I. Jang, F. Arvin, and A. Lanzon, “A Decentralized Cluster Formation Containment Framework for Multirobot Systems,” *IEEE Trans. Robot.*, vol. 37, no. 6, pp. 1936–1955, 2021, doi: 10.1109/TRO.2021.3071615.
- [117] N. B. Cruz, N. Nedjah, and L. de Macedo Mourelle, “Robust distributed spatial clustering for swarm robotic based systems,” *Appl. Soft Comput.*, vol. 57, pp. 727–737, 2017, doi: 10.1016/j.asoc.2016.06.002.
- [118] G. Di Caro, F. Ducatelle, and L. M. Gambardella, “A fully distributed communication-based approach for spatial clustering in robotic swarms,” *Proc. 2nd Auton. Robot. Multirobot Syst. Work. (ARMS), Affil. with 11th Int. Conf. Auton. Agents Multiagent Syst.*, pp. 153–171, 2012.
- [119] A. Dutta, U. Vladimir, A. Asai, and E. and Czarnecki, “Coalition Formation for Multi-Robot Task Allocation via Correlation Clustering,” *Cybern. Syst.*, vol. 50, no. 8, pp. 711–728, Nov. 2019, doi: 10.1080/01969722.2019.1677334.
- [120] A. Prorok, A. Bahr, and A. Martinoli, “Low-cost collaborative localization for large-scale multi-robot systems,” in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 4236–4241, doi: 10.1109/ICRA.2012.6225016.
- [121] J. Chen, C. Du, Y. Zhang, P. Han, and W. Wei, “A Clustering-Based Coverage Path Planning Method for Autonomous Heterogeneous UAVs,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 12, pp. 25546–25556, 2022, doi: 10.1109/TITS.2021.3066240.
- [122] O. Arslan, D. P. Guralnik, and D. E. Koditschek, “Coordinated Robot Navigation via Hierarchical Clustering,” *IEEE Trans. Robot.*, vol. 32, no. 2, pp. 352–371, 2016, doi: 10.1109/TRO.2016.2524018.
- [123] O. Arslan, “Clustering-Based Robot Navigation and Control,” 2016.

- [124] K. Zhang, E. G. Collins, and D. Shi, “Centralized and distributed task allocation in multi-robot teams via a stochastic clustering auction,” *ACM Trans. Auton. Adapt. Syst.*, vol. 7, no. 2, Jul. 2012, doi: 10.1145/2240166.2240171.
- [125] N. Nedjah, L. M. Ribeiro, and L. de Macedo Mourelle, “Communication optimization for efficient dynamic task allocation in swarm robotics,” *Appl. Soft Comput.*, vol. 105, p. 107297, 2021, doi: 10.1016/j.asoc.2021.107297.
- [126] S. Opiyo, C. Okinda, J. Zhou, E. Mwangi, and N. Makange, “Medial axis-based machine-vision system for orchard robot navigation,” *Comput. Electron. Agric.*, vol. 185, p. 106153, 2021, doi: 10.1016/j.compag.2021.106153.
- [127] H. Rong, A. Ramirez-Serrano, L. Guan, and Y. Gao, “Image Object Extraction Based on Semantic Detection and Improved K-Means Algorithm,” *IEEE Access*, vol. 8, pp. 171129–171139, 2020, doi: 10.1109/ACCESS.2020.3025193.
- [128] H. Zhang and Q. Peng, “PSO and K-means-based semantic segmentation toward agricultural products,” *Futur. Gener. Comput. Syst.*, vol. 126, pp. 82–87, 2022, doi: 10.1016/j.future.2021.06.059.
- [129] J. Li and F. Yang, “Task assignment strategy for multi-robot based on improved Grey Wolf Optimizer,” *J. Ambient Intell. Humaniz. Comput.*, vol. 11, no. 12, pp. 6319–6335, 2020, doi: 10.1007/s12652-020-02224-3.
- [130] F. Janati, F. Abdollahi, S. S. Ghidary, M. Jannatifar, J. Baltes, and S. Sadeghnejad, “Multi-robot Task Allocation Using Clustering Method BT - Robot Intelligence Technology and Applications 4,” 2017, pp. 233–247.
- [131] M. Elango, S. Nachiappan, and M. K. Tiwari, “Balancing task allocation in multi-robot systems using K-means clustering and auction based mechanisms,” *Expert Syst. Appl.*, vol. 38, no. 6, pp. 6486–6491, 2011, doi: 10.1016/j.eswa.2010.11.097.
- [132] E. Murugappan, S. Nachiappan, R. Shams, G. Mark, and H. K. and Chan, “Performance analysis of clustering methods for balanced multi-robot task

- allocations,” *Int. J. Prod. Res.*, vol. 60, no. 14, pp. 4576–4591, Jul. 2022, doi: 10.1080/00207543.2021.1955994.
- [133] H. Zhang, L. Bai, J. Qi, and Y. Xiao, “Area Coverage of Swarm Robotics Based on Anti-Flocking Framework with Dynamical Clustering,” in *2022 IEEE International Conference on Unmanned Systems (ICUS)*, 2022, pp. 209–213, doi: 10.1109/ICUS55513.2022.9986539.
- [134] A. Solanas and M. A. Garcia, “Coordinated multi-robot exploration through unsupervised clustering of unknown space,” in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, 2004, vol. 1, pp. 717–721 vol.1, doi: 10.1109/IROS.2004.1389437.
- [135] Y. Yuan, P. Yang, H. Jiang, and T. Shi, “A Multi-Robot Task Allocation Method Based on the Synergy of the K-Means++ Algorithm and the Particle Swarm Algorithm,” *Biomimetics*, vol. 9, no. 11, 2024, doi: 10.3390/biomimetics9110694.
- [136] W. C. Wang and R. Chen, “An Effective Way to Large-Scale Robot-Path-Planning Using a Hybrid Approach of Pre-Clustering and Greedy Heuristic,” *Appl. Artif. Intell.*, vol. 34, no. 14, pp. 1159–1175, Dec. 2020, doi: 10.1080/08839514.2020.1824094.
- [137] Nur Aira Abd Rahman, Khairul Salleh Mohamed Sahari, Nasri A Hamid, and Yew Cheong Hou, “A coverage path planning approach for autonomous radiation mapping with a mobile robot,” *Int. J. Adv. Robot. Syst.*, vol. 19, no. 4, p. 17298806221116484, Jul. 2022, doi: 10.1177/17298806221116483.
- [138] S. Sharma, R. Tiwari, A. Shukla, and J. Yadav, “Canopy Clustering Based Multi Robot Area Exploration,” *IFAC Proc. Vol.*, vol. 47, no. 1, pp. 505–510, 2014, doi: 10.3182/20140313-3-IN-3024.00253.
- [139] N. J. Cho, S. H. Lee, and I. H. Suh, “Modeling and evaluating Gaussian mixture model based on motion granularity,” *Intell. Serv. Robot.*, vol. 9, no. 2, pp. 123–

- 139, 2016, doi: 10.1007/s11370-015-0190-1.
- [140] J. Yan, Y. Wu, K. Ji, C. Cheng, and Y. Zheng, “A novel trajectory learning method for robotic arms based on Gaussian Mixture Model and k-value selection algorithm,” *PLoS One*, vol. 20, no. 2 February, pp. 1–19, 2025, doi: 10.1371/journal.pone.0318403.
 - [141] Y. Q. Wang, Y. D. Hu, S. El Zaatari, W. D. Li, and Y. Zhou, “Optimised Learning from Demonstrations for Collaborative Robots,” *Robot. Comput. Integr. Manuf.*, vol. 71, p. 102169, 2021, doi: 10.1016/j.rcim.2021.102169.
 - [142] C. Ye, J. Yang, and H. Ding, “Bagging for Gaussian mixture regression in robot learning from demonstration,” *J. Intell. Manuf.*, vol. 33, no. 3, pp. 867–879, 2022, doi: 10.1007/s10845-020-01686-8.
 - [143] S. H. Lee, I. H. Suh, S. Calinon, and R. Johansson, “Autonomous framework for segmenting robot trajectories of manipulation task,” *Auton. Robots*, vol. 38, no. 2, pp. 107–141, 2015, doi: 10.1007/s10514-014-9397-9.
 - [144] J. Lou, “Crawling robot manipulator tracking based on gaussian mixture model of machine vision,” *Neural Comput. Appl.*, vol. 34, no. 9, pp. 6683–6693, 2022, doi: 10.1007/s00521-021-06063-x.
 - [145] T. M. N. U. Akhund *et al.*, “IoST-Enabled Robotic Arm Control and Abnormality Prediction Using Minimal Flex Sensors and Gaussian Mixture Models,” *IEEE Access*, vol. 12, pp. 45265–45278, 2024, doi: 10.1109/ACCESS.2024.3380360.
 - [146] A. Mora, A. Prados, A. Mendez, R. Barber, and S. Garrido, “Sensor Fusion for Social Navigation on a Mobile Robot Based on Fast Marching Square and Gaussian Mixture Model,” *Sensors*, vol. 22, no. 22, 2022, doi: 10.3390/s22228728.
 - [147] Y. Yuan, J. Liu, W. Chi, G. Chen, and L. Sun, “A Gaussian Mixture Model Based Fast Motion Planning Method Through Online Environmental Feature Learning,” *IEEE Trans. Ind. Electron.*, vol. 70, no. 4, pp. 3955–3965, 2023,

doi: 10.1109/TIE.2022.3177758.

- [148] P. Yao, Q. Zhu, and R. Zhao, “Gaussian Mixture Model and Self-Organizing Map Neural-Network-Based Coverage for Target Search in Curve-Shape Area,” *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 3971–3983, 2022, doi: 10.1109/TCYB.2020.3019255.
- [149] G. J. McLachlan and S. Rathnayake, “On the number of components in a Gaussian mixture model,” *WIREs Data Min. Knowl. Discov.*, vol. 4, no. 5, pp. 341–355, Sep. 2014, doi: 10.1002/widm.1135.
- [150] M. Deisenroth, A. Faisal, and C. Ong, *Mathematics for Machine Learning*. Cambridge University Press, 2020.