

새로운 휴리스틱 기반 청소 및 분리수거 작업 경로 생성 기법

김민규*, 이승환**

A New Heuristic Approach for Generating Efficient Paths for Cleaning and Recycling Tasks

Min-Gyu Kim*, Seung-Hwan Lee**

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터사업의 연구결과 (IITP-2024-RS-2024-00437190, 50%), 2025년도 산업통상부 재원으로 해양수산과학기술진흥원의 지원(20210630, 극한지개발 및 탐사용 협동이동체 시스템 기술개발, 50%)과 2025년도 경상북도 지역혁신중심 대학지원체계(RISE)-(특성화 대학)의 지원을 받아 수행된 결과임.

요약

기존 로봇 청소기는 객체를 구분하지 않고 면적 기반 경로에 의존해 다양한 물체 수거에 한계가 있다. 본 연구는 복합 환경에서 청소와 분리수거를 동시에 수행하도록 문제를 정의하고, 이동 거리와 회전 각도를 함께 고려한 커버리지 경로 계획을 제안한다. 제안 방법은 대표적인 휴리스틱 기법인 유전 알고리즘(GA)과 A* 알고리즘을 결합하여 경로를 생성한다. 구체적으로 시작점과 객체 위치, 장애물, 종류별 목적지를 입력받아 시작점에서 가장 가까운 객체의 종류를 판별한 뒤, 그 객체들을 먼저 수거한다. 이후 장애물을 고려하면서 나머지 객체를 지정된 목적지로 운반하는 경로를 계획하며 이 과정에서 이동 거리와 회전 각도를 최소화한다. 제안 방법의 타당성을 검증하기 위해 두 가지 시나리오 기반의 시뮬레이션과 실제 실험을 수행하였다. 실험 결과 제안 방법은 기존 방식에 비해 경로 길이를 최대 1.78%, 회전 각도를 최대 5.30%까지 줄였다.

Abstract

Existing robotic vacuums follow area-based paths without object recognition, limiting effective debris collection. This study proposes a path planning method for combined cleaning and sorting, considering both travel distance and rotation angle for efficient operation. The proposed method first identifies the type of the nearest object to the start location and uses a genetic algorithm to determine an efficient visitation sequence. Each object type is then collected and transported to its designated destination via paths computed by the A* algorithm that avoid obstacles. The fitness of a path is evaluated based on total travel distance and cumulative rotation angles, aiming to minimize both. We validated the effectiveness of the proposed method through two scenario-based simulations and real-world experiments. The results show that our method reduces path length by up to 1.78% and rotation angle by up to 5.30% compared to conventional approaches, improving overall performance.

Keywords

A* algorithm, genetic algorithm, heuristic-based coverage path planning, distance-angle efficiency

* 국립금오공과대학교 기계공학과

- ORCID: <https://orcid.org/0009-0004-7205-163X>

** 국립금오공과대학교 제어및로봇전공 교수(교신저자)

- ORCID: <https://orcid.org/0000-0001-8426-975X>

• Received: May 07, 2025, Revised: May 30, 2025, Accepted: Jun. 02, 2025

• Corresponding Author: Seung-Hwan Lee

Dept. of Engineering, Kumoh National Institute of Technology,
61 Daehak-ro (yangho-dong), Gumi, Gyeongbuk, [39177] Korea
Tel.: +82-54-478-7458, Email: leesh@kumoh.ac.kr

I. 서론

스마트홈 확산과 인공지능 기술의 발전으로 로봇 청소기 시장은 빠르게 성장하고 있으며, 제조·물류·서비스 전반에 걸쳐 로봇 수요도 지속적으로 증가하고 있다. 특히 코로나19 이후 자동화와 비대면 서비스 수요가 급증하면서 관련 시장은 두 자릿수 성장세를 이어가고 있으며[1], 자동 경로 설정과 장애물 회피 등 고도화된 기능의 청소 로봇이 잇따라 등장하고 있다[2]. 그러나 실제 주거 및 공공 환경에서는 머리카락, 먼지 외에도 다양한 크기와 형태의 버려진 소형 물체(예: 종이, 캔 등)가 혼재되어 있으며, 기존 로봇 청소기는 일정한 패턴이나 면적 기반 커버리지 방식에 의존하기 때문에, 물체의 유형이나 위치 관계를 고려한 효율적인 수거 경로 계획에는 한계가 있다[3][4]. 이에 따라 우선순위 판단, 경로 통합, 이동 효율성 측면에서 개선이 요구된다.

본 논문에서는 이러한 복합 환경에서 청소와 분리수거를 동시에 수행할 수 있는 차세대 로봇 청소기를 위한 경로 문제를 정의하였고, 이동 거리와 방향 전환을 동시에 고려한 효율적인 커버리지 경로 계획(Coverage path planning) 기법을 제안한다. 제안한 방법은 먼저, 시작점과 환경 내 객체, 목적지, 장애물의 위치를 정의한 후, 시작점에서 가장 가까운 객체를 선택하여 종류를 판별하고 목적지까지의 경로를 생성한다. 이후 객체 간 경유 순서는 유전 알고리즘(GA, Genetic Algorithm)[5]을 통해 유동적으로 결정되며, 각 경유지 간의 세부 경로는 장애물 회피를 고려하여 A* 알고리즘[6]을 활용해 산출된다. 이러한 절차를 통해 생성된 경로는 이동 거리와 회전 각도를 순차적으로 최소화하여 하나의 통합 경로로 구성되며, 최종적으로 적합도 평가를 통해 거리와 각도 측면에서 효율적인 커버리지 경로가 선정된다.

따라서, 본 연구는 단순 면적 기반 방식에서 객체 중심의 인지형 경로 계획으로 확장함으로써 다양한 청소 및 분리수거 대상 물체를 효과적으로 수거할 수 있도록 설계되었다. 제안된 알고리즘의 효율성과 실제 환경에서의 적용 가능성은 두 가지 시나리오 기반 시뮬레이션 및 실제 실험을 통해 검증하였다.

II. 관련연구

2.1 최단 경로 탐색

최단 경로 탐색 문제는 두 노드 사이의 거리, 시간, 에너지 등의 비용을 최소화하는 경로 탐색 문제를 의미하며, 그래프 이론과 인공지능 분야에서 Dijkstra, Floyd, A* 등의 연구가 활발히 진행중이다[7]. Dijkstra 알고리즘은 양수 가중치를 갖는 그래프에서 트리 구조로 경로를 확장하며 최단 경로를 탐색한다. 그러나 장애물이 많은 환경에서는 탐색 영역이 증가하여 계산량이 커지는 단점이 있다[8]. Floyd 알고리즘은 모든 노드 쌍 간 최단 경로를 계산할 수 있지만, 연산량이 많아 대규모 그래프에는 비효율적이다[9]. A*는 시작점에서 현재 노드까지의 실제 비용과 목표 지점까지의 예상 비용의 합을 기준으로 경로를 탐색하는 휴리스틱 기반 기법이다[6]. 이때, 평가 비용은 두 요소의 합으로 계산되며, 이를 통해 탐색의 효율성과 정확도를 동시에 확보할 수 있다. Open 리스트 증가 문제를 해결하기 위해 Dijkstra와 A*를 결합한 하이브리드 방식도 제안되었으며, 탐색 효율과 계산량 감소에 기여하였다[10]. 또한, 기존 조선소 블록 운송 과정에서 발생하는 경로 손상·비정형 재고 배치·작업 변동 대응을 위한 연구에서는 Floyd, Dijkstra, A* 알고리즘을 비교·분석하고 상황별 선택 기준을 제시했으며, 경로 재설계 및 빈 차량의 실시간 최적 배정 기법을 제안하였다[11]. 연구 결과, Floyd는 3중 중첩 반복 구조로 높은 계산 비용이 발생했다. A*는 탐색 중 손상 노드를 실시간 회피해 일대일 경로 탐색에 효율적인 반면, Dijkstra는 사전 손상 정보를 반영함으로써 다대다 경로 탐색에서 가장 우수한 성능을 보였다. 따라서 전체 노드 간 모든 경로를 산출하는 문제에는 Floyd, 다대다 경로 탐색에는 Dijkstra, 특정 지점 간 일대일 경로 탐색에는 목적지 인접 노드를 우선 탐색하는 A*를 적용하는 것이 적합하다. 본 연구에서는 관련 선행 연구를 근거로, 경유지 개수와 관계없이 장애물을 회피하며 두 노드 간 최단 경로를 산출해야 하는 경로 계획에 A* 알고리즘을 채택하였다.

2.2 커버리지 경로 계획

미로와 같은 대규모 네트워크에서 효과적인 탈출 경로를 찾는 문제는 동적 상황과 높은 복잡도로 인해 해결이 어려운 경우가 많다. 특히, 커버리지 경로 계획에서 경유지 수가 증가하면 탐색 비용이 팩토리얼 수준으로 급격히 증가하여 경로 최적화가 어려워진다. 이를 해결하기 위해 개미 군집 최적화(ACO, Ant colony optimization)[12]와 GA[5] 같은 휴리스틱 기법이 널리 활용되고 있다. ACO는 개미들이 경로에 남긴 정보를 기반으로 최적 경로를 찾는 방법으로, 확률적 선택과 페로몬 업데이트를 반복하여 경로를 점진적으로 최적화한다. 국소 최적해 탐색에 효과적이지만, 전체 탐색 속도가 느리고 전역 최적해 도달에는 한계가 있다[12]. 반면, GA는 자연 선택과 유전 원리에 기반한 전역 최적화 기법이다. 문제 공간에서 가능한 다양한 해를 반영하도록 초기 개체군을 무작위로 생성한 후, 각 개체의 성능을 평가하여 적합도를 산출한다. 이후, 적합도가 높은 개체를 우선 선택하고 교차 연산을 통해 유전 정보를 교환하여 새로운 해를 도출하며, 돌연변이 연산을 통해 다양성을 확보하는 과정을 반복함으로써 점진적으로 최적 해에 수렴한다[13]-[16]. 특히, [17]에서는 대학 강의 스케줄링 문제에 휴리스틱 알고리즘인 GA와 ACO를 적용하여 동일한 제약 조건 하에 반복 실험 결과, GA가 계산 시간과 메모리 사용 면에서 더 우수한 성능을 보였다. 본 연구에서는 다수의 경유지 수가 존재하는 환경이며, 수렴 속도가 저하되는 특성을 가지는 ACO 기반 방법에 비해 병렬 연산 구조를 활용하여 다양한 해를 동시에 평가하고, 넓은 탐색 공간에서 효과적으로 최적 해를 도출할 수 있는 GA[18]가 커버리지 경로 계획 산출에 적합한 알고리즘으로 판단된다.

2.3 기존 경로 계획

기존 경로 계획은 외판원 순회 문제(TSP, Traveling Sales Problem)를 기반한 사례의 접근법을 적용하여 여러 지점을 방문하면서 거리를 최소화한다.

첫 번째로, 차량 경로 문제(Vehicle routing problem)에서는 여러 경유지를 거쳐 차고지로 복귀할 때 전체 이동 거리와 시간을 최소화하기 위한 목적성을 가지며, NP-hard의 최적 해를 찾는 데 높은 계산 비용이 발생하는 특성 때문에, 주로 근사 해법이 사용된다[19]. 국내 택배 시스템은 인접 이웃 알고리즘(NNA, Nearest Neighbor Algorithm)을 활용하여 출발지에서 가까운 고객부터 방문하는 방식으로 경로를 설정하며, 각 영업소별 구역을 분할한 후, 운행 종료 시 출발지로 복귀하는 방식으로 운영된다[20]. 또 다른 연구에서는 NNA 기법을 적용하여 가장 가까운 객체를 우선 방문함으로써 드론의 이동 방향을 최적화하고, 데이터 전달 경로의 효율성을 높인다. 특히, 무선 센서 네트워크(Wireless sensor network)는 전력 소모 문제 해결을 위해 무인 항공기(Unmanned aerial vehicle) 기반 데이터 수집 방식을 적용하며, 출발지에서 최단 거리 노드를 우선 경유한 후 전체 경로를 최적화하여 데이터 전달 속도를 향상시키고 경로 최적화 문제의 해결방안을 제시하였다[21].

앞선 두 연구에서는 NNA를 활용해 출발지에서 가장 가까운 객체로 이동한 후 나머지 경로의 순서를 최적화하여 경로를 계획하였다. 그러나 첫 번째 경유지가 고정됨에 따라 전체 경로 최적화 과정이 비효율적일 수 있다. 특히 분리수거 작업은 객체별 수거 지점이 상이하기 때문에, 단일 경로만으로는 모든 객체 수거에 한계가 있다. 따라서 종류별 수거 위치를 설정하고, 목적지에 도달할 때마다 다음 수거 대상을 탐색하며, 객체 유형을 고려한 경로를 재설정해야 한다. 이에 본 연구는 시작점에서 가장 가까운 객체의 종류를 식별하고, 해당 종류의 모든 객체를 우선적으로 수거하는 전략을 채택하였다. 이후 각 유형에 맞는 지정 지점까지의 경로를 계획하면서 이동 거리와 회전 각도를 최소화하도록 하였다. 시작점의 첫 대상 선택에는 단일 대상에 즉시 접근하는 NNA 방식을 적용하되, 이후의 경로 계획에는 제안한 방법을 적용하여 기존 방식의 한계를 분석하고 성능을 평가하고자 한다.

III. 제안한 방법(구조설명)

휴리스틱 기반 로봇 청소기의 청소 및 분리수거 경로 효율화를 위한 전체적인 순서도는 그림 1에 나타났다. 먼저, 시작점과 객체, 장애물, 그리고 종류별 목적지 좌표의 환경을 입력값으로 설정한다. 이후, 유클리드 거리 계산식[22]을 이용해 시작점에서 가장 가까운 객체 선정 및 종류를 판별하고, 선택된 종류를 제외한 나머지 객체들을 장애물로 인식하여 경로를 계획한다. 기존 경로 계획은 시작점에서 가장 가까운 객체로 이동한 후, 그 위치를 새로운 시작점으로 설정하고 동일 객체들을 수거한 후 목적지로 이동하는 경로를 계획한다. 이때, 대표적인 휴리스틱 기법인 GA[20]를 활용하며, 객체의 위치는 2차원 좌표(x, y)로 표현되어 염색체의 유전자로 인코딩된다. 반면, 제안한 방법은 가장 가까운 객체를 포함한 동일 종류의 객체들을 대상으로 최소 이동 거리와 회전 각도를 만족하는 효율적인 커버리지 경로를 생성한다.

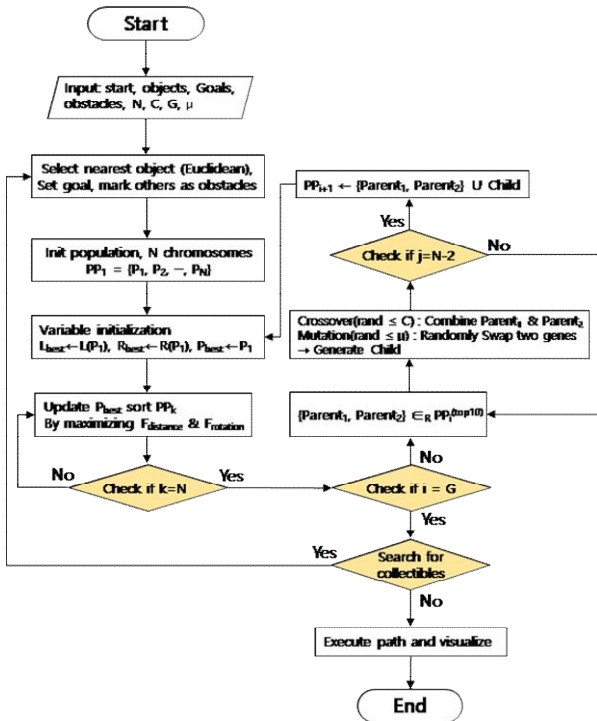


그림 1. 제안한 경로 효율화 방법의 순서도
Fig. 1. Proposed path efficiency flowchart

초기 개체군은 객체의 모든 위치와 경유 순서를 포함하는 N 개의 경로(염색체)로 구성된다. 먼저 이

동 거리와 회전 각도를 기준으로 우선순위 기반 적합도를 평가하여 상위 염색체를 선별 및 정렬하고, 이어서 교차와 돌연변이 연산을 통해 새로운 자식 개체를 생성한다. 이때 엘리트 보존 전략을 적용해 우수 염색체가 손실되지 않도록 유지하며, 지정된 최대 세대에 도달할 때까지 이 과정을 반복한다. 그 결과 이동 거리와 회전 각도가 최소화된 경유지 순서를 지닌 염색체가 최적 해로 결정된다. i 번째 산출된 염색체인 경로 P_i 는 n 개의 노드에 대해 시작점 $p_{i,1}$ 에서 출발하여 모든 경유지를 거쳐 목적지 $p_{i,n}$ 에 도달하는 경로로 정의된다. 이때, 각 구간 거리는 인접한 노드 간의 유클리드 거리로, 아래의 식 (1)과 같이 계산된다.

$$D(p_{i,j}, p_{i,j+1}) = \sqrt{(x_{i,j+1} - x_{i,j})^2 + (y_{i,j+1} - y_{i,j})^2} \quad (1)$$

$x_{i,j}$ 와 $y_{i,j}$ 는 i 번째 산출된 경로 P_i 의 j 번째 경유지의 2차원 좌표를 의미한다. P_i 의 전체 거리 $L(P_i)$ 는 구간 거리 $D(\cdot)$ 의 누적합으로 식 (2)와 같이 계산된다.

$$L(P_i) = \sum_{j=1}^{n-1} D(p_{i,j}, p_{i,j+1}) \quad (2)$$

이를 바탕으로, 거리 측면에서 전체 이동 거리를 최소화하기 위해, 전체 경로 길이를 반영한 평가 요소 $F_{distance}(L(P_i))$ 를 식 (3)과 같이 계산된다.

$$F_{distance}(L(P_i)) = \frac{1}{L(P_i)} \quad (3)$$

이 거리 적합도 함수는 전체 거리가 작을수록 높은 값이 된다. 본 연구에서는 거리 측면에서의 적합도 평가뿐만 아니라 경로상의 회전 각도를 평가하기 위해 경로 P_i 의 누적 이동 방향 정보 $R(P_i)$ 를 제안한다. $R(P_i)$ 는 연속된 노드의 방향 차인 $\Delta\theta_{i,j}$ 의 절댓값의 합으로 식 (4)와 같이 계산된다.

$$R(P_i) = \sum_{j=2}^{n-1} |\Delta\theta_{i,j}| \quad (4)$$

i 번째 산출된 경로의 j 번째 노드의 방향 차의 절대치 $|\Delta\theta_{i,j}|$ 는 $|\theta_{i,j} - \theta_{i,j-1}|$ 로 계산되며, j 번째 노드에서 $j+1$ 번째 노드까지의 방향은 식 (5)와 같이 계산된다.

$$\theta_{i,j} = \arctan2(y_{i,j+1} - y_{i,j}, x_{i,j+1} - x_{i,j}) \quad (5)$$

최종적으로, 에너지 효율성을 고려한 누적 회전 각도를 최소화하기 위해, 전체 회전 각도를 반영한 평가 요소 $F_{rotation}(R(P_i))$ 는 누적 회전 각도가 작을수록 큰 값이 되며 식 (6)과 같이 계산된다.

$$F_{rotation}(R(P_i)) = \frac{1}{R(P_i)} \quad (6)$$

제안한 방법의 핵심인 이동 거리와 회전 각도를 고려한 효율적인 커버리지 경로 생성 과정을 표 1에 기술하였다. 입력으로는 객체 수 n , 시작점 (*start*)과 목적지(*goal*), 교차율 C , 돌연변이율 μ , 최대 세대 수 G , 개체군 크기 N 이 사용된다. 초기 개체군 PP_1 은 $n-2$ 개의 무작위 경유 순서를 포함하는 N 개의 염색체로 구성된다. 각 세대에서는 개체군 PP_i 의 염색체들을 이동 거리와 회전 각도를 기준으로 적합도를 평가하고, 이를 정렬하여 다음 세대 생성을 위한 기준으로 활용한다.

표 1. 제안한 경로 효율화 방법의 의사코드

Table 1. Pseudocode of proposed path efficiency method

Algorithm 1: Proposed efficiency path planning
Input: $PP_1 = \{P_1, P_2, \dots, P_N\}$, n , <i>start</i> , <i>goal</i> , μ , C , G
Output: P_{best}
1. for $i = 1$ to $G-1$ do
2. $PP_i, P_{best} \leftarrow \text{Sort_Population}(PP_i)$
3. $\text{Child_List} \leftarrow \emptyset$
4. for $j = 1$ to $N-2$ do
5. $\{\text{Parent}_1, \text{Parent}_2\} \leftarrow \text{Randomly select } PP_i^{(top10)}$
6. $\text{Child} \leftarrow \text{CrossMut}(\text{Parent}_1, \text{Parent}_2, C, \mu)$
7. $\text{Child_List} \leftarrow \text{Child_List} \cup \{\text{Child}\}$
8. end for
9. $PP_{i+1} \leftarrow \{P_1, P_2\} \cup \text{Child_List}$
10. end for
11. $PP_G, P_{best} \leftarrow \text{Sort_Population}(PP_G)$
return P_{best}

정렬된 개체군 PP_i 에서 상위 10개 개체 중 무작위로 두 부모를 선택한 후, 교차와 돌연변이 연산을 반복하여 $N-2$ 개의 자식 개체를 생성하며, 이를 Child_List 에 저장한다. 이 과정에서는 C 확률로 부모 유전자를 교차하고, μ 확률로 자식 개체 내 유전자를 무작위로 교환한다. PP_i 의 상위 두 개체 $\{P_1, P_2\}$ 와 Child_List 개체들을 합친 총 N 개의 염색체로 다음 세대의 개체군 PP_{i+1} 을 구성한다.

최대 세대 수(G)에 도달하면, 개체군 PP_G 에서 가장 효율적인 염색체 P_{best} 를 최종 해로 반환한다.

매 세대마다 개체군 내 모든 염색체의 이동 거리와 회전 각도를 평가하여 최상위 경로 P_{best} 를 갱신하고, 성능 순으로 정렬하는 과정을 표 2에 기술하였다. 초기 세대 개체군 PP_1 에서 P_1 은 무작위로 생성된 경로를 나타내지만, 이후 세대 개체군 $PP_k (k \geq 2)$ 에서는 이전 세대의 최우수 염색체가 P_1 으로 선정되며, P_1 의 거리 L_1 과 회전각 R_1 을 각각 L_{best} 와 R_{best} 로 초기화하여 모든 개체 비교의 기준으로 활용한다. 반복문에서는 각 개체의 이동 거리 L_i 가 작은 경로 P_{best} 를 우선 갱신하고, 동일할 경우 회전 각도 R_i 가 더 작은 경로 P_{best} 를 갱신한다. 이 과정을 총 N 회 반복하여, 이동 거리 L_{best} 가 최소이고 그중 회전 각도 R_{best} 가 최소인 염색체 P_{best} 를 도출하며, 최종 세대 G 에서는 갱신된 최상위 염색체 P_{best} 가 최종 해로 결정된다.

표 2. 개체군 염색체의 평가 및 정렬 알고리즘

Table 2. Chromosome evaluation and ranking algorithm

Algorithm 2: Sort_Population(PP_k)
Input: $PP_k = \{P_1, P_2, \dots, P_N\}$
Output: PP_k, P_{best}
1. $L_{best} \leftarrow L(P_1), R_{best} \leftarrow R(P_1), P_{best} \leftarrow P_1$
2. for $k = 2$ to N do
3. if $F_{distance}(L(P_k)) > F_{distance}(L_{best})$ then
4. $P_{best} \leftarrow P_k$
5. $L_{best} \leftarrow L(P_k)$
6. $R_{best} \leftarrow R(P_k)$
7. else if $F_{distance}(L(P_k)) == F_{distance}(L_{best})$ then
8. if $F_{rotation}(R(P_k)) > F_{rotation}(R_{best})$ then
9. $P_{best} \leftarrow P_k$
10. $R_{best} \leftarrow R(P_k)$
11. end if
12. end if
13. end for
14. $PP_k \leftarrow \text{Sort}(PP_k, \text{by}=[F_{distance}, F_{rotation}])$
return PP_k, P_{best}

현 세대 개체군 PP_k 의 염색체들은 평가 요소 $F_{distance}$ 와 $F_{rotation}$ 에 따라 거리 기준 내림차순(동일 시 각도 기준 내림차순)으로 정렬된다. 이렇게 재구성된 염색체 집단을 다음 세대 개체군에 반영함으로써, 점진적으로 우수한 해를 도출해 낸다.

교차 및 돌연변이 연산을 반복하여 자식 개체를 생성하는 과정을 표 3에 기술하였다. 정렬된 i 세대 개체군 PP_i 에서 상위 10개의 염색체 중 무작위로 두 개체($Parent_1, Parent_2$)를 부모로 지정하고, 교차 확률 C 에 따라 $Parent_1$ 에서 임의의 구간 $[s, e]$ 을 설정하여, 유전자 서열($Segment$)을 정의한다. $Parent_2$ 에서는 유전자 서열에 포함되지 않는 유전자들을 원래 순서를 유지한 채 $Remaining$ 으로 추출하며, 이를 유전자 서열의 뒤에 이어붙여 자식 개체 $Child$ 를 생성한다.

교차 확률 미충족 시, 부모 개체인 $Parent_1$ 또는 $Parent_2$ 중 하나를 자식 개체로 그대로 선택한다. 이후, 자식 개체는 돌연변이 확률 μ 에 따라 유전적 변형을 거친다. 이때, 염색체 $Child$ 내 임의의 두 지점 a, b 를 선택하고, $Child_a$ 와 $Child_b$ 의 경유지 순서를 서로 교환($Swap$)함으로써 유전적 다양성을 확보한다.

표 3. 교차, 돌연변이 기반 자식 개체 생성 알고리즘
Table 3. Crossover, mutation operation for chromosome

<p>Algorithm 3: CrossMut($Parent_1, Parent_2, C, \mu$)</p> <p>Input: $Parent_1, Parent_2, C, \mu$</p> <p>Output: Child</p> <ol style="list-style-type: none"> 1. if $Random(0,1) \leq C$ then 2. Randomly select s, e such that $1 \leq s < e \leq Parent_1$ 3. $Segment \leftarrow Parent_1[s : e]$ 4. $Remaining \leftarrow [p \in Parent_2 \mid p \notin Segment]$ 5. $Child \leftarrow Segment \parallel Remaining$ 6. else 7. if $Random(0,1) < 0.5$ then 8. $Child \leftarrow Parent_1$ 9. else 10. $Child \leftarrow Parent_2$ 11. end if 12. end if 13. if $Random(0,1) < \mu$ then 14. Randomly select a, b such that $1 \leq a, b \leq Child$ 15. $Swap(Child_a, Child_b)$ 16. end if <p>return Child</p>

매 반복마다 하나의 자식 개체가 $Child_List$ 에 저장되며, 기존 개체군 PP_i 내 상위 두 개체 P_1 과 P_2 는 다음 세대 PP_{i+1} 에 보존된다. 따라서, 다음 세대는 이 두 개체와 새로 생성된 자식 개체들로 구성되며, 이러한 세대 교체 과정이 $G-1$ 회 반복된다.

최종 세대 G 에서 구성된 N 개의 모든 염색체를 이동 거리와 회전 각도를 기준으로 평가하여 개체군 PP_G 에서 가장 우수한 염색체 P_1 을 선별한다. 이를 통해 전체 세대를 거쳐 도출된 효율적인 커버리지 경로 P_{best} 를 최종 결과로 반환한다.

IV. 시뮬레이션 또는 실험

경로 탐색은 상하좌우뿐만 아니라 대각선 이동도 허용하도록 설계되었으며, 시작 시 초기 방향은 상단을, 종류별 목적지 도착 후에는 좌측을 향하도록 설정하였다. 또한, GA 파라미터는 실험을 통해 가장 적합한 성능을 보이는 값으로 선정하여 표 4에 제시하였다.

표 4. 적용 파라미터
Table 4. Applied parameters

Parameter	G	N	C	μ
Value	1000	50	1.00	0.05

4.1 시나리오 1

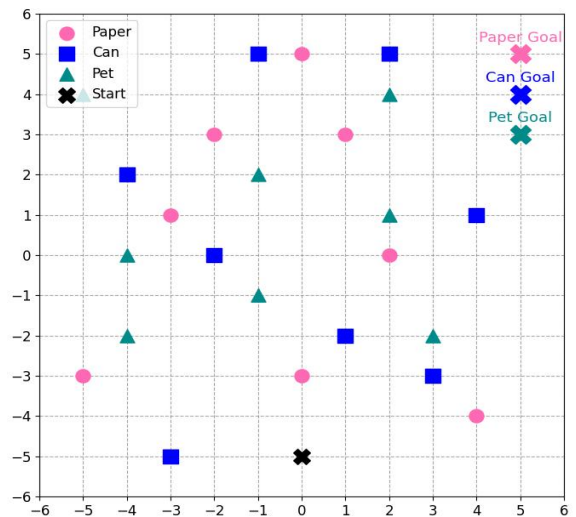


그림 2. 시나리오 1 환경
Fig. 2. Environment for scenario 1

시나리오 1의 환경은 12×12 공간에 종이(Paper), 캔(Can), 펫병(Pet)을 8개씩 배치하고, 종이는 분홍색 원, 캔은 남색 네모, 펫병은 녹색 세모로 구분했다. 시작점은 (0, -5)에 검은색 X 표시이며, 종류별 목적지는 우측 상단에 동일 모양의 Paper goal,

Can goal, Pet goal로 나타났다. 전체 환경 구성은 그림 2에 제시하였다.

그림 3은 시나리오 1 환경에서의 경로 산출 시각화를 나타내며, 큰 변화가 있는 부분은 동일한 색상의 사각형으로 표시하여 비교를 분명히 하였다.

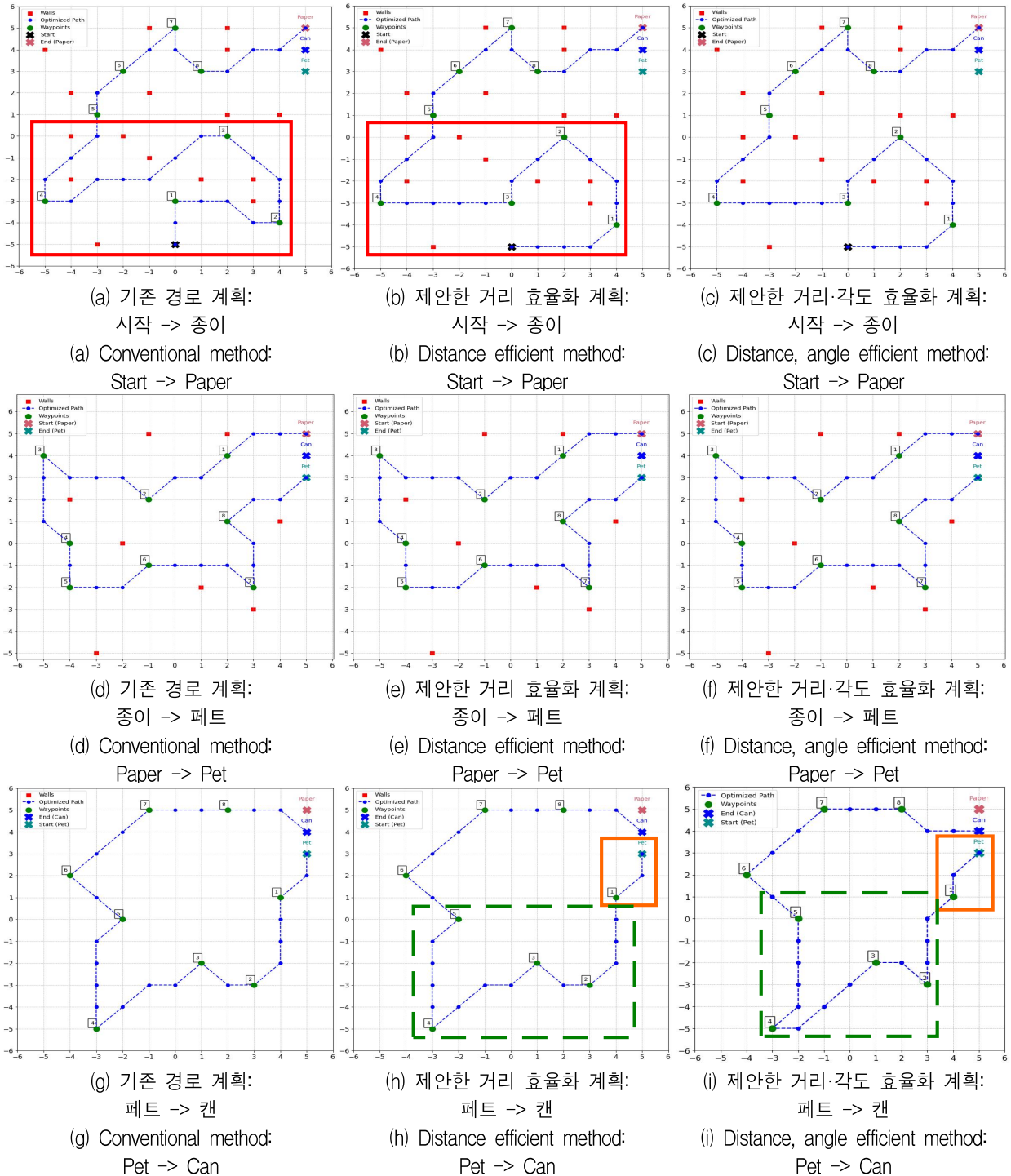


그림 3. 시나리오 1 환경에서의 경로 산출 시각화

Fig. 3. Visualization of path calculation in scenario 1 environment

거리 효율화 방식을 적용하면서 (a)에서 (b)로 변경됨에 따라 첫 번째 경유지가 (4, -4)로 조정되었으며, 4번째 경유지까지의 이동은 대각 이동이 6회에서 5회로, 회전 횟수는 10회에서 7회으로 감소하였다. 또한, (c)에서도 동일하게 효율적인 경로가 산출되었다.

거리와 각도 효율화를 적용한 결과, 첫 번째 경유지 설정뿐 아니라 경로상의 회전 각도도 효율성에 영향을 끼친다. 5번째 경유지까지의 이동에서 (g)와 (h)는 동일한 경로를 보이며, (i)로 변경 시 회전 횟수가 11회에서 10회로 감소하였다. 또한, 출발지에서 하단 45도 방향으로 이동하는 등 불필요한 회전을 최소화하고 경로 전반의 회전 각도를 줄이는 효과를 보이며, 산출 결과를 표 5에 제시하였다.

기존 경로 계획과 거리 효율화 방법을 비교한 결과, 첫 번째 경유지가 동일할 경우 결과는 일치했다. 그러나, 전체 경로에서는 기존 경로의 이동 거리 518.7 pixel, 회전 각도 55.75 rad에 비해, 거리 효율화 방법 적용 시 511.6 pixel, 54.18 rad로 감소하는 차이를 보였다. 특히, 종이 수거 경로에서 첫 번째 경유지 좌표가 (4, -4)로 변경되면서 이동 거리는 7.1 pixel, 회전 각도는 1.57 rad 감소하였다. 이는 거리 효율화 방법이 이동 거리와 회전 각도를 함께 감소시킨 결과로 해석된다.

제안한 거리-각도 효율화 적용 결과, 총 이동 거리는 511.6 pixel, 회전 각도는 53.4 rad로 산출됐으며, 거리 효율화 방법 대비 누적 회전각이 0.78 rad 감소했다. 캔 수거 경로는 출발 시 하단 45도 이동으로 초기 회전 각을 줄이고 이후 경유지에서도 회전 최소화를 고려하여 누적 회전 각도가 감소했다. 반면, 페트병 수거 경로는 모든 방법에서 동일한 결과를 보였다. 거리와 회전 각도를 동시에 고려한 경로 계획을 적용한 결과, 로봇청소기의 이동 경로는 직선적인 유도가 이루어졌다. 다만, 이동 방향이 45도 단위 회전으로 제한되어 회전 각도의 변화는 크지 않았다.

표 5. 시나리오 1 환경에서의 경로 검증 결과
Table 5. Path validation results in the scenario 1 environment

Sequence	NNA method		Proposed method (Distance efficient)		Proposed method (Distance, angle efficient)	
	Length(pixel)	Rotation(rad)	Length(pixel)	Rotation(rad)	Length(pixel)	Rotation(rad)
Start -> Paper	184.00	19.63	176.90	18.06	176.90	18.06
Paper -> Pet	167.80	19.63	167.80	19.63	167.80	19.63
Pet -> Can	166.92	16.49	166.92	16.49	166.92	15.71
Sum	518.70	55.75	511.60	54.18	511.60	53.40

4.2 시나리오 2

시나리오 2의 환경은 20 x 20 공간에 종이 (Paper), 캔(Can), 페트병(Pet), 유리(Glass), 금속 (Metal) 객체를 각각 10개씩 배치하고, 종이는 분홍색 원, 캔은 남색 네모, 페트병은 청록색 세모, 유리는 별 모양, 금속은 진한 회색 마름모로 구분하였다. 초기 시작점은 (0, -9)에 검은색 X로 표시되며, 종류별 목적지는 지도 우측 상단에 동일한 모양과 함께 Paper goal, Can goal, Pet goal, Glass goal, Metal goal로 나타내며, 직사각형 장애물을 추가하여 전체 환경 구성을 그림 4에 제시하였다.

시나리오 1에서는 지정된 객체 종류를 제외한 나머지 객체들을 장애물로 인식하여 경로를 산출하는 것이 비교적 용이하다. 그러나 실제 가정집과 같은 실내 공간에서는 분리수거가 필요한 특정 객체 외에도, 경로 이행 시 공통적으로 장애물로 인식되는 요소들이 존재하여 경로 산출에 영향을 미칠 수 있다.

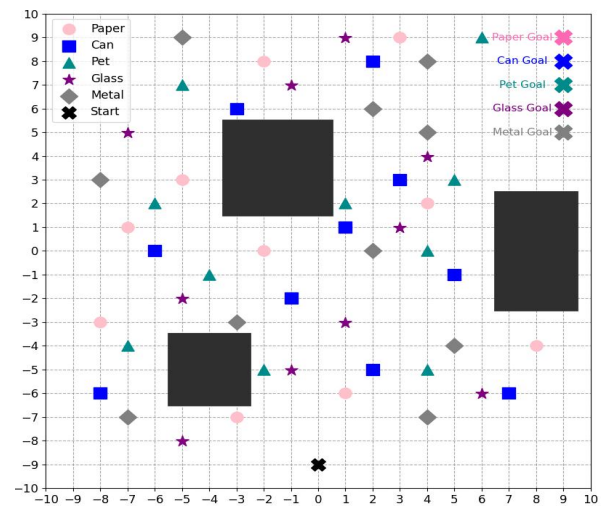


그림 4. 시나리오 2 환경
Fig. 4. Environment for scenario 2

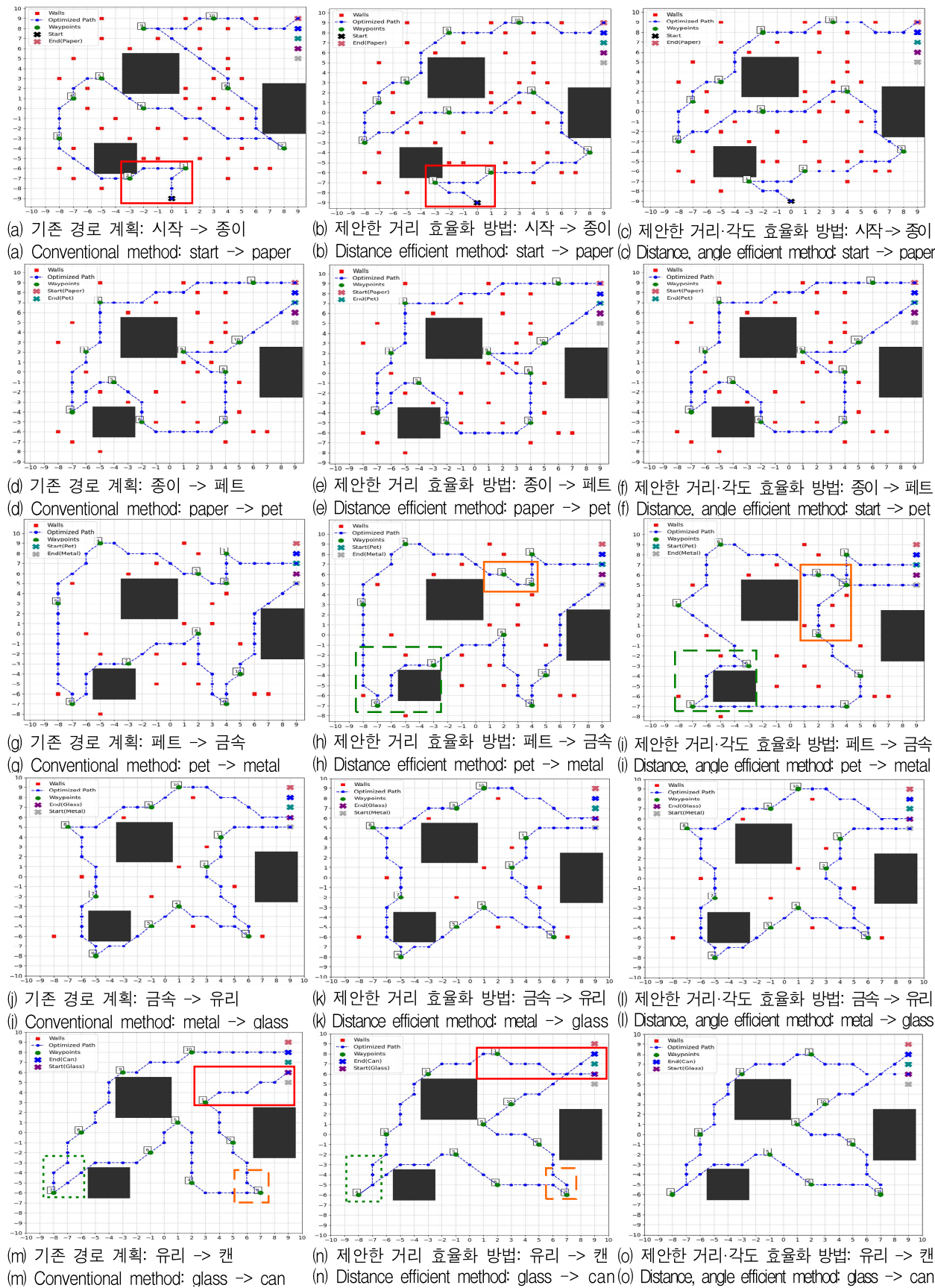


그림 5. 시나리오 2 환경에서의 경로 산출 시각화

Fig. 5. Visualization of path calculation in the scenario 2 environment

시나리오 2에서는 장애물을 회피하는 과정이 전체 경로 길이와 회전 각도에 영향을 줄 수 있다. 또한, 실제 환경에서는 충돌 회피와 경로 재탐색과 같은 동적인 변화에 대응해야 하므로, 장애물 회피 능력과 센서 정확도가 중요한 요소가 된다. 따라서, 다양한 환경에서 효율적인 경로를 도출할 수 있도록 제안한 방법의 실현 가능성과 범용성을 검증하였다.

시나리오 2 환경에서 기존 방법과 경로 효율화 방법별 성능을 표 6에 나타내었다. 우선 거리 효율화 방법을 적용했을 때, 종이 수거 경로의 첫 번째 경유지는 기존 (4, -4)에서 (-3, -7)로 변경되면서 이에 따라 직선 이동 횟수는 22회에서 25회로, 회전 횟수는 23회에서 26회로 증가했으며, 대각 이동은 23회에서 19회로 감소했다. 다만, 이 개선은 9번째 경유지까지만 적용되었으며 이후 구간은 기존 경로와 동일하게 유지되었다. 반면, 캔 수거 경로의 경우 첫 번째 경유지를 (2, 8)로 조정한 결과 직선 이동 횟수가 38회에서 28회로 감소하고, 대각 이동은 22회에서 28회로 증가했으며, 회전 횟수는 34회에서 31회로 감소한 시각적 결과를 보였다. 이에 따라 전체 경로 길이는 1,677.80 pixel에서 1,651.95 pixel로 25.85 pixel 감소했으며, 총 회전 각도는 148.45 rad에서 146.09 rad으로 2.36 rad 감소하여, 이동 거리와 회전 효율이 모두 향상되었다. 다른 수거 경로들은 거리 효율화 전후에 차이를 보이지 않았다. 한편, 거리 및 회전 복합 효율화 방법을 적용하면 전체 이동 거리는 동일하게 유지되는 가운데 직선 이동 42회, 대각 이동 21회를 일관되게 관리하고 회전 횟수를 32회에서 27회로 줄여, 총 회전 각도를 146.09 rad에서 140.59 rad로 5.50 rad만큼 절감한 결과를 보였다.

따라서, 시나리오 1 환경에서 제안한 방법의 유효성을 확인한 후, 장애물과 다수의 종류와 객체가 혼재된 시나리오 2 환경에서 시뮬레이션을 수행했다. 그 결과 제안한 방법은 기존 방법에 비해 이동 거리와 회전 각도 모두 향상시켰다. 또한, 장애물을 효과적으로 회피하여 경유지 간 최단 경로 탐색이 가능함을 확인하였다. 이러한 시뮬레이션 검증 결과가 실제 주행 시에도 동일한 성능이 재현되는지 추가 검증이 필요하다.

4.3 실제 실험

앞선 시뮬레이션 검증에서는 장애물의 유무에 따라 환경을 구성한 후, 제안한 경로 효율화 방법을 적용하여 이동 거리와 회전량이 경로 계획에 미치는 영향성을 분석하였다. 실제 실험은 앞선 시나리오 1 환경을 구성하였으며, 이를 통해 제안한 방법의 효율성을 평가하였다. 그림 6[13]은 로봇의 기구학 구조를 나타내며, X축은 전진 거리, Y축은 좌우 이동을 의미하며, 회전 각도는 θ 로 표현된다. 로봇은 시작점에서 정면을, 목적지 도달 후에는 좌측을 향하도록 초기값을 설정했으며, 이러한 설정을 반영하여 전체 경로를 구성했다[13].

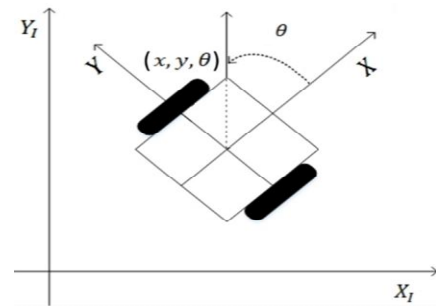


그림 6. 차동 이륜 구동 로봇 기구학 구조[13]

Fig. 6. Kinematic structure of differential drive dobot[13]

표 6. 시나리오 2 환경에서의 경로 검증 결과

Table 6. Path validation results in the scenario 2 environment

Sequence	NNA method		Proposed method (Distance efficient)		Proposed method (Distance, angle efficient)	
	Length(pixel)	Rotation(rad)	Length(pixel)	Rotation(rad)	Length(pixel)	Rotation(rad)
Start → Paper	338.85	29.06	320.55	29.06	320.55	29.06
Paper → Pet	310.20	22.78	310.20	22.78	310.20	22.78
Pet → Metal	358.50	32.20	358.50	32.20	358.50	26.70
Metal → Glass	324.70	32.99	324.70	32.99	324.70	32.99
Glass → Can	345.55	31.42	338.00	29.06	338.00	29.06
Sum	1677.80	148.45	1651.95	146.09	1651.95	140.59

실험에는 Yujin robot의 kobuki 모델을 사용했으며, 해당 로봇은 차동 이륜 구동형 구조로 하단부에 두 개의 DC 모터가 탑재되고, 전면부에 T자 형태의 구조물을 결합했다. 크기는 지름 351.5 mm, 높이가 124.8 mm로 설계되었으며, 상단에 2D LIDAR 센서(LD06)을 부착하였다. 로봇의 위치 추정 및 장애물 인식을 위해 SLAM[23] 기법을 적용함으로써, 실시간 위치 정보를 바탕으로 제안한 방법을 적용하여 각 객체 종류별 목적지까지의 효율적인 이동 경로를 계획하고 실제로 수행하였다.

또한, 객체의 종류와 위치 정보는 상단 벽면에 설치된 CCTV로 수집한 좌표를 활용했으며, 앞서 구성한 시나리오 1 환경을 기반으로, 그림 7과 같이 실험을 수행하였다. 각 객체 종류별 1개씩 바운딩 박스와 라벨링을 표시하고, 종류별 목적지는 직선으로 표시하여 구분하였다. 실험 수행 결과는 누적 거리와 회전 각도를 수치적으로 분석하여 표 7에 제시하였다. 기존 경로 계획 대비 경로 효율화 방법은 이동 거리와 회전 각도가 각각 1.95 m, 0.29 rad 감소하였으며, 추가로 각도 효율화 방법은 회전 각도가 62.28 rad로 1.58 rad만큼 더 줄어드는 결과를 보였다. 이는 제안한 방법이 이동 거리뿐만 아니라 회전 각도까지 함께 고려함으로써 에너지 소비 절감에도 효과적임을 확인하였다.

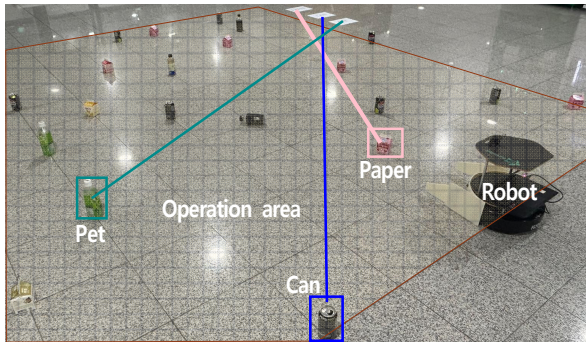


그림 7. 실험 환경
Fig. 7. Experiment environment

표 7. 실제 실험 환경에서의 경로 수행 결과
Table 7. Path execution results in the real experiment

Sequence	NNA method		Proposed method (Distance efficient)		Proposed method (Distance, angle efficient)	
	Length(m)	Rotation(rad)	Length(m)	Rotation(rad)	Length(m)	Rotation(rad)
Sum	109.51	64.15	107.56	63.86	107.56	62.28

V. 결 론

제안한 방법은 시작점에서 유클리드 거리 기반으로 가장 가까운 객체를 선정한 후, 해당 객체의 종류를 판별하여 동일 유형의 수거 계획을 수립한다. 경유지 순서는 GA를, 장애물 우회를 고려한 경유지 간 이동은 A*를 적용하였다. 전체 경로는 적합도 평가를 기반으로 최소 이동 거리와 회전 각도를 고려한 효율적인 커버리지 경로를 제시하였다.

기존 경로 방법, 제안한 거리 효율화 방법, 제안한 거리-각도 효율화 방법을 두 가지 시나리오 기반 시뮬레이션을 수행하여 제안한 방법의 성능 비교와 함께 타당성을 검증하였으며, 실제 실험은 앞선 시나리오 1을 구성하여 제안한 방법의 적용 가능성을 평가하였다. 전체적으로 얻어진 결과는 아래와 같다.

1) 시나리오 1 환경에서의 검증 결과, 기존 경로 계획 대비 이동 거리가 1.37%, 회전 각도가 4.20% 줄어들었으며, 거리 효율화 방법과 비교했을 때 회전 각도가 1.44% 감소한 효과를 보였다.

2) 또한, 시나리오 2 환경에서는 기존 경로 계획 대비 이동 거리가 1.54%, 회전 각도가 5.30% 감소하였고, 거리 효율화 방법과 비교했을 때 회전 각도가 3.76% 줄어드는 결과를 확인하였다.

3) 실제 실험 결과, 도달 허용 오차와 회전 각도 오차의 영향으로 일부 곡선 형태의 이동이 발생하였으나, 기존 경로 계획 대비 이동 거리가 1.78% 감소하였고, 회전 각도는 2.90% 감소했다. 거리 효율화 경로와 비교했을 때 회전 각도가 2.48% 감소하였다.

본 연구에서는 제안한 경로 효율화 기법의 영향성을 탐색하였다. 협소하고 복잡한 환경에서도 불필요한 경로 우회를 줄이고, 이동 거리와 회전 각도를 순차적으로 최소화함으로써 주행 효율성과 에너지 절감을 극대화할 수 있다. 단순 면적 기반 패턴에서 객체의 위치와 유형을 고려한 상황 인지형 경로 계획의 실현 가능성을 제시하였다.

향후에는 제안한 방법의 적용 범위를 상업 공간이나 복층 구조 등 복잡한 환경으로 확장 적용하여, 동적 장애물과 실시간 경로 수정이 요구되는 상황에서도 경로 효율성을 검증할 예정이다.

References

- [1] Hyundai Motor Group, <https://www.hyundai.co.kr/story/CONT0000000000001671>. [accessed: Feb. 28, 2025]
- [2] H. H. Lee, Y. M. Yim, T. H. Min, S. H. Kim, G. H. Lee, Y. H. Choi, and H. A. Lee, "Obstacle recognition using a camera and an infrared sensor for robots to climb up walls", Proc. 23rd Annual Conf. of KIPS, Seoul, Korea, pp. 25-27, Apr. 2016. <https://doi.org/10.3745/PKIPS.y2016m04a.25>.
- [3] Q. Huang, "Towards Indoor Suctionable Object Classification and Recycling: Developing a Lightweight AI Model for Robot Vacuum Cleaners", Appl. Sci, Vol. 13, No. 18, pp. 10031, Sep. 2023. <https://doi.org/10.3390/app131810031>.
- [4] S. H. Ha, I. C. Choe, H. S. Kim, and H. T. Jeon, "Collision-Avoidance and Optimal Path Planning of Autonomous Mobile Robot using Soft-Computing", Journal of Korean Institute of Intelligent Systems, Vol. 20, No. 2, pp. 195-201, Apr. 2010. <https://doi.org/10.5391/JKIIS.2010.20.2.195>.
- [5] A. Akkaya and C. Közkurt, "Genetic Algorithm Research: A Bibliometric Analysis", Pioneer and Innovative Studies in Engineering, All Sciences Academy Design, pp. 309-331, Dec. 2024.
- [6] T. K. Whangbo, "Efficient Bidirectional Search Algorithm for Optimal Route", Journal of Korea Multimedia Society, Vol. 5, No. 6, pp. 745-752, Dec. 2002.
- [7] G. S. Choi and K. B. Woo, "An Optimal and Genetic Route Search Algorithm for Intelligent Route Guidance System", Journal of Institute of Control, Robotics and Systems, Vol. 3, No. 2, pp. 156-161, Apr. 1997.
- [8] S. U. Lee, "A Real-time Shortest Path Search for Navigation Based on Traveling Time Using the Minimum Spanning Tree", Journal of KIIT, Vol. 12, No. 8, pp. 1-8, Aug. 2014. <https://doi.org/10.14801/kitr.2014.12.8.1>.
- [9] H. V. Chung, T. W. Kim, and Y. M. Chung, "Design of Emergency Evacuation Guiding System with Serially Connected Multi-channel Speakers", Journal of the Institute of Electronics Engineers of Korea SP, Vol. 48, No. 4, pp. 142-152, Jul. 2011.
- [10] Y. H. Lee and S. W. Kim, "A Hybrid Search Method of A* and Dijkstra Algorithms to Find Minimal Path Lengths for Navigation Route Planning", Journal of the Institute of Electronics and Information Engineers, Vol. 51, No. 10, pp. 109-117, Oct. 2014. <http://dx.doi.org/10.5573/ieie.2014.51.10.109>.
- [11] J. H. Moon, W. S. Ruy, and J. H. Cha, "Comparison of Optimal Path Algorithms and Implementation of Block Transporter Planning System", Journal of the Society of Naval Architects of Korea, Vol. 53, No. 2, pp. 115-126, Apr. 2016. <https://doi.org/10.3744/SNAK.2016.53.2.115>.
- [12] G. H. Kim and J. J. Chae, "Airline Disruption Management Using Ant Colony Optimization Algorithm with Re-timing Strategy", Journal of Society of Korea Industrial and Systems Engineering, Vol. 40, No. 2, pp. 13-21, May 2017. <https://doi.org/10.11627/jkise.2017.40.2.013>.
- [13] Y. K. Choi and J. H. Park, "Predictive Control based on Genetic Algorithm for Mobile Robots with Constraints", Journal of the Korea Institute of Information and Communication Engineering, Vol. 22, No. 1, pp. 9-16, Jan. 2018. <https://doi.org/10.6109/jkiice.2018.22.1.9>.
- [14] H. S. Son, J. H. Park, and Y. K. Choi, "Predictive Control for Mobile Robots Using Genetic Algorithms", Journal of the Korea Institute of Information and Communication Engineering, Vol. 21, No. 4, pp. 698-707, Apr. 2017. <https://doi.org/10.6109/jkiice.2017.21.4.698>.
- [15] Y. K. Choi and J. H. Park, "Control Gain Optimization for Mobile Robots Using Neural Networks

- and Genetic Algorithms", Journal of the Korea Institute of Information and Communication Engineering, Vol. 20, No. 4, pp. 698-706, Apr. 2016. <https://doi.org/10.6109/jkiice.2016.20.4.698>.
- [16] J. Zhu and D. Pan, "Improved Genetic Algorithm for Solving Robot Path Planning Based on Grid Maps", Mathematics, Vol. 12, No. 24, pp. 4017, Dec. 2024. <https://doi.org/10.3390/math12244017>.
- [17] I. A. Ashari, M. A. Muslim, and A. Alamsyah, "Comparison Performance of Genetic Algorithm and Ant Colony Optimization in Course Scheduling Optimizing", Scientific Journal of Informatics, Vol. 3, No. 2, pp. 51-60, Nov. 2016. <http://dx.doi.org/10.15294/sji.v3i2.7911>.
- [18] S. H. Kang, M. S. Choi, M. A. Jung, and S. R. Lee, "A Pareto Ant Colony Optimization Algorithm for Application-Specific Routing in Wireless Sensor & Actor Networks", The Journal of Korean Institute of Communications and Information Sciences, Vol. 36, No. 4, pp. 346-353, Apr. 2011. <https://doi.org/10.7840/KICS.2011.36B.4.346>.
- [19] S. J. Park, L. Hua, and G. H. Lee, "Routing Optimization of Mobile Library Using Vehicle Routing Problem (VRP)", Journal of the Korean Cartographic Association, Vol. 22, No. 3, pp. 25-36, Dec. 2022. <https://doi.org/10.16879/jkca.2022.22.3.025>.
- [20] H. S. Lee and K. H. Yoo, "Application of Ant System Algorithm on Parcels Delivery Service in Korea", Journal of the Korean Society of Transportation, Vol. 23, No. 4, pp. 81-91, Sep. 2005.
- [21] T. S. Alemayehu and J. H. Kim, "Efficient Nearest Neighbor Heuristic TSP Algorithms for Reducing Data Acquisition Latency of UAV Relay WSN", Wireless Pers Commun, Vol. 95, No. 3, pp. 3271-3285, Aug. 2017. <https://doi.org/10.1007/s11277-017-3994-9>.
- [22] H. I. Kim, "Path Planning for Cleaning Robots Using Virtual Map", Journal of the KSCI, Vol. 14, No. 11, pp. 31-40, Nov. 2009. <https://doi.org/10.9708/jksci.2009.14.11.031>.
- [23] H. D. Jung and G. W. Kim, "GG-SLAM: An Object-Level SLAM System Using Gaussian

Grouping and Splating", The Journal of Korea Robotics Society, Vol. 20 No. 1, pp. 84-93, Nov. 2024. <https://doi.org/10.7746/jkros.2025.20.1.084>.

저자소개

김민규 (Min-Gyu Kim)



2025년 2월 : 국립금오공과대학교
전자공학부(학사)

2025년 3월 ~ 현재 :

국립금오공과대학교
기계공학과(항공기계전자융합전공)
석사과정

관심분야 : 커버리지 경로 계획,

CBM, Digital Twin

이승환 (Seung-Hwan Lee)



2015년 8월 : 서울대학교
전기컴퓨터학부(공학박사)

2015년 9월 ~ 2018년 2월 :
삼성전자 생산기술연구소
책임연구원

2018년 3월 ~ 2023년 3월 :

국립금오공과대학교 전자공학부

조교수

2023년 4월 ~ 현재 : 국립금오공과대학교 전자공학부

부교수

관심분야 : 가중 로봇 SLAM, 다중 로봇 커버리지 경로 계획