

물류 및 생산시스템의 효율 향상을 위한 RRT-GRID 알고리즘 기반 자율주행로봇 경로 계획

김성현¹, 권익현^{2*}

¹인제대학교 일반대학원 AI융합로봇학과, ²인제대학교 산업경영공학과

Path Planning of Autonomous Mobile Robots Based on RRT-GRID Algorithm for Efficiency Improvement in Logistics and Production Systems

Sung Hyun Kim¹, Ick-Hyun Kwon^{2*}

¹Department of AI Convergence Robotics, Graduate School, Inje University

²Department of Industrial and Management Engineering, Inje University

요약 정보통신기술과 인공지능 기술의 발달로 물류 및 생산시스템은 자동화에 대한 수요가 증가하면서 자율주행로봇이 중요한 역할을 수행하고 있다. 자율주행로봇은 제조 공장 및 물류센터와 같은 다양한 환경에서 작업을 수행하며, 물류 및 생산 작업의 효율성과 생산성을 향상하는 데 기여한다. 따라서, 자율주행로봇의 경로 계획은 이러한 시스템의 최적화에 있어서 매우 중요한 문제이다. 본 논문에서는 물류 및 생산시스템의 최적화를 위해 자율주행로봇에서 주로 사용되는 경로 계획 알고리즘인 RRT(Rapidly-Exploring Random Tree) 알고리즘을 기반으로 하여 그리드 및 경유 노드를 활용한 RRT-GRID 알고리즘을 제시하였다. 제안된 RRT-GRID 알고리즘은 기존 RRT 알고리즘이 취약한 성능을 나타내는 복잡하고 넓은 영역의 환경에서 경로 최적화 및 계산 소요 시간 단축을 주요 목적으로 한다. 본 논문에서는 제안된 RRT-GRID 알고리즘을 실제 물류 및 생산시스템 환경을 반영한 시뮬레이션 모형을 통해 검증하였고 복잡한 환경에서 RRT-GRID 알고리즘이 우수한 성능을 나타냄을 확인하였다. 본 논문에서 제안한 방법론과 실험 결과를 토대로 물류 및 생산시스템에서 자율주행로봇의 경로 계획의 중요성을 확인하였고, 자율주행로봇의 경로 계획 최적화를 통해 물류 및 제조 산업의 효율성과 생산성 향상에 기여할 수 있을 것으로 기대한다.

Abstract With the rapid advancement of information and communication technology and artificial intelligence, there is a growing demand for automation in logistics and production systems, where autonomous mobile robots play a crucial role. These robots operate in diverse environments, including manufacturing plants and logistics centers, and significantly contribute to enhancing the efficiency and productivity of logistics and production operations. Consequently, the optimization of path planning for autonomous mobile robots has become a vital concern in these systems. This paper proposes an algorithm named RRT-GRID, which builds upon the widely used rapidly exploring random tree algorithm in autonomous mobile robots to optimize logistics and production systems. The RRT-GRID algorithm leverages the integration of grids and waypoints to overcome the limitations of the conventional RRT algorithm in complex and extensive environments. Its primary objectives include path optimization and reduction of computation time. The effectiveness of the proposed RRT-GRID algorithm is demonstrated through simulation models that replicate real-world logistics and production system environments, which showcase its performance in complex scenarios. Based on the methodology proposed and experimental results in this paper, we confirmed the significance of path planning for autonomous robots in logistics and production systems. Furthermore, it is expected that optimizing the path planning of autonomous robots will contribute to enhancing the efficiency and productivity of logistics and manufacturing industries.

Keywords : Autonomous Mobile Robot, Logistics and Production Systems, Path Planning, RRT, Simulation

본 논문은 제1저자의 석사학위 논문을 수정, 보완한 연구임.

*Corresponding Author : Ick-Hyun Kwon(Inje Univ.)

email: ikwon@inje.ac.kr

Received July 28, 2023

Revised August 30, 2023

Accepted October 6, 2023

Published October 31, 2023

1. 서론

정보통신기술과 인공지능 기술의 발달로 인해 다양한 제조 및 물류 시스템에서 자동화 설비와 자동화 로봇에 대한 수요가 증가하고 있다[1]. 특히 자율주행로봇 (Autonomous Mobile Robot, 이하 AMR)과 무인운반차(Automated Guided Vehicle, 이하 AGV) 등이 중요한 역할을 수행하고 있다. 이 가운데 공간상의 제약과 사람, 기계의 이동이 많아서 충돌 위험성이 높은 물류 및 제조 현장에서는 주로 AMR이 사용되고 있다[2].

AMR은 주변 환경을 센서를 통해 인식하여 주변 환경의 상황 변화에 따라 실시간으로 경로를 계획하거나 변경한다. 따라서 동적인 환경에서 높은 유연성을 가지고 있다는 장점이 있다. 특히 다품종 소량생산 시스템은 소품종 대량생산 방식보다 요구되는 자재의 종류가 많으며 각각의 셀마다 다른 제품을 생산함으로 인하여 공장의 물류 흐름이 복잡하기 때문에, 정해진 경로를 따라 움직이는 AGV보다 경로 탐색 알고리즘을 통하여 스스로 경로를 찾는 AMR을 사용하는 경우가 일반적이다.

AMR은 경로 탐색을 통해 복잡한 환경에서 운영하기 적합하지만, 다른 AMR 또는 사람 또는 장애물이 많고 이동 거리가 멀다면 경로 계획의 소요 시간이 길어지거나 비효율적인 경로로 이동할 수 있다는 단점을 가지고 있다. 복잡하거나 넓은 환경에서 경로 탐색을 위해서는 AMR마다 고성능의 연산 시스템과 시각 센서가 필요하다는 단점이 있지만, 최근에는 엣지 컴퓨팅, 클라우드 컴퓨팅과 같은 원격 클라우드 시스템을 통한 중앙 제어 시스템으로 AMR이 개별적으로 경로 탐색을 하는 것이 아닌 AMR을 통제하는 중앙 컴퓨터가 AMR의 경로 탐색 및 운영을 관리하는 접근법이 등장하였다. 원격 클라우드 시스템을 통해 AMR의 경로 탐색을 위한 부하를 줄이고 경로 탐색 시간 또한 단축되는 등 AMR에 고성능의 연산 시스템을 직접 탑재하지 않더라도 운용 가능한 기반이 구축되어 AMR의 보급을 촉진하고 있다.

AMR은 현재 다양한 환경에서 효율적으로 사용하기 위해 폭넓게 연구되고 있으며 AMR에 사용되는 로봇 몸체, 센서, 인코더, 구동부, 통신 모듈 등과 같은 하드웨어적인 분야뿐만 아니라 로봇 위치인식(Localization), 주행 환경 인식, 경로 계획, 원격 제어 기술 등과 같은 소프트웨어적인 분야 또한 많은 연구가 진행 중에 있다[3,4].

이중 경로 계획 기술은 자율주행에서 가장 중요한 기술로 오랜 기간 연구되고 있다. 경로 계획은 샘플링 기반 알고리즘과 탐색(Searching) 기반 알고리즘으로 나눌

수 있는데 샘플링 기반 알고리즘은 복잡한 환경에서 경로를 찾을 수 있지만 찾은 경로가 최적 경로임을 보장하지는 않는다. 탐색 기반 알고리즘은 경로에 대한 정보를 상세히 파악하여 최적 경로를 찾을 수 있으나 대규모 환경에서 사용하기 힘들다는 단점이 있다.

따라서 실제 AMR에는 샘플링 기반 알고리즘 중 동적인 환경에서 장애물 회피가 가능한 RRT(Rapidly-exploring Random Tree) 기반 알고리즘이 주로 사용되고 있다. 하지만 RRT 알고리즘은 장애물이 많은 복잡한 환경 또는 넓은 영역에서는 무작위 샘플링 방식으로 인하여 경로 탐색에 많은 시간이 소요되며, 알고리즘을 통해 도출된 경로가 최적 경로를 보장하지 않는다는 단점을 가진다[5].

Table 1. Classification of Prior Research

Research Area		Related Research
Reduction of Calculation Time	Algorithm Time Reduction	Islam et al.[6], Lee et al.[7], Connell and La[8], Park[9]
Route Optimization	Performance Improvement	Seif and Oskoei[10], Otte and Frazzoli[11], Gammell et al.[12], Tak et al.[13], Kang et al.[14]
	Application of other Functions and Algorithms	Kuffner and LaValle[15], Perez et al.[16], Li et al.[17], Hong et al.[18]

Table 1은 RRT 알고리즘을 개선하기 위한 다양한 기존 연구를 주제별로 분류한 표이다. RRT 알고리즘의 단점을 보완하거나 개선하기 위하여 RRT 알고리즘의 무작위 샘플링 방법을 개선하거나 타 함수 또는 알고리즘 등을 통해 RRT 알고리즘을 개선하고자 하는 접근방법을 적용한 RRT-Connect, RRT*, LQR(Linear Quadratic Regulation)-RRT*, RRT*-Smart, Informed RRT* 등의 다양한 알고리즘이 연구되었으며 테스트 환경 내에서 경로 길이를 단축하거나 계산 소요 시간을 단축하는 결과를 보였다. 하지만 실제 물류센터 및 생산시스템과 같은 복잡하고 넓은 공간을 고려한 환경에서 RRT 알고리즘을 적용하거나 개선하는 연구는 부족하였다.

본 연구의 목적은 복잡하고 넓은 영역을 포함하는 다양한 물류센터 및 생산시스템 내에서 AMR의 활용성을 개선하기 위한 새로운 알고리즘을 개발하는 것이다. 이를 위해 기존의 RRT 알고리즘에 그리드(격자) 및 경유노드를 설정하는 방안을 접목하여 넓은 영역에서 탐색

시간을 단축하고, RRT 알고리즘의 샘플링 방식을 개선하여 경로 최적화 문제를 보완하고자 한다. 실제 물류센터 및 생산시스템의 형태를 반영한 시뮬레이션 모형을 설계하고, 이를 기반으로 기존 알고리즘과의 비교·분석을 통해 제안하는 알고리즘의 성능을 검증하고자 한다. 본 논문에서 제안하는 알고리즘에 기반한 자율주행로봇의 경로 계획을 통해 물류 및 제조 산업 현장에서의 효율성과 생산성 향상에 기여할 수 있을 것으로 기대한다.

2. 본론

2.1 RRT 알고리즘

RRT 알고리즘은 2D 또는 3D 환경에서 두 지점 사이의 경로를 찾기 위한 경로 계획에 널리 사용되는 샘플링 기반 알고리즘으로 공간 전역에 무작위 샘플링 방식으로 시작 위치와 목표 위치 사이의 환경 및 가능한 경로를 나타내는 트리와 같은 구조를 구축한다.

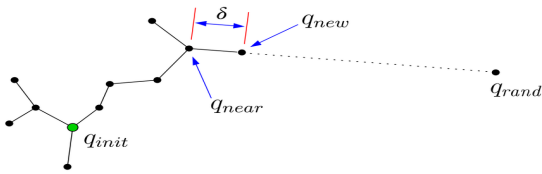


Fig. 1. Tree Construction Process of RRT[15]

Fig. 1은 RRT 알고리즘의 트리 구축 과정을 나타낸 것이다. 시작 시 시작 위치를 나타내는 단일 노드(q_{init})가 트리에 배치된다. 이러한 환경에서 알고리즘은 무작위 샘플(q_{rand})을 반복적으로 생성하고 트리에서 샘플에 가장 가까운 노드(q_{near})를 찾아 가장 가까운 노드에서 샘플을 향해 트리를 확장하여 장애물을 만나거나 최대 거리에 도달하는 지점에서 트리의 새 노드(q_{new})를 만든다. 이러한 과정을 반복하여 목표 위치에 대한 경로를 찾거나 무한 루프 방지를 위해 매우 큰 값으로 설정된 최대 반복 횟수에 도달할 때까지 이를 반복한다. RRT 알고리즘은 다양하고 복잡한 환경에서도 목적지까지 가는 이동 경로를 찾을 수 있으나 찾은 경로를 수정하는 과정이 없고 무작위 샘플링을 통하여 경로를 생성하므로 최적 경로임을 보장하지는 않으며 장애물이 많은 복잡한 환경에서는 경로 탐색 시간이 오래 소요되는 단점이 있다. 이러한 현상은 장애물이 많거나 경로의 길이가 길어질수록 더욱 문제가 될 수 있다.

2.2 GRID 기반 RRT 알고리즘 개선 방안

본 연구에서는 RRT 알고리즘을 기반으로 출발 노드와 도착 노드 사이 영역을 그리드로 나누고 나누어진 그리드를 바탕으로 경유 노드를 설정하여 복잡하거나 넓은 영역에서 효율적인 알고리즘을 개발하고자 한다. 그리드는 연속적인 영역을 일정한 크기의 작은 셀(Cell) 또는 요소(Element)로 분할하여 이루어진 격자 모양의 형태이다. 그리드로 영역을 분할하고 그리드의 꼭짓점 가운데 3개의 임시 경유 노드를 설정 후 각 노드로의 도달 가능 여부를 확인한다. 필요하다면 4개의 임시 경유 노드를 추가로 설정하고 도달 가능 여부를 확인한다. 임시 경유 노드 3개의 도달 가능 여부를 확인하여 도달 가능한 임시 경유 노드는 경유 노드로 확정하고, 만약 장애물로 인해 도달 불가능한 임시 경유 노드는 삭제한다. 경유 노드가 확정된다면 출발 노드와 경유 노드들의 경로 길이 및 소요 시간을 계산 후, 경유 노드 간의 경로 길이 및 소요 시간을 계산한다. 다음으로 모든 경유 노드에서 도착 노드까지의 경로 길이 및 소요 시간을 계산한다. 마지막으로 이렇게 얻어진 값들을 입력값으로 하여 최단 경로 문제를 해결하는 데 주로 사용되는 Dijkstra 알고리즘[19]을 통하여 출발 노드로부터 도착 노드까지의 최적 이동 경로를 구한다. 이처럼 본 연구에서 제안하는 알고리즘은 기존 RRT 알고리즘을 기반으로 GRID를 통해 영역을 나누고 나누는 영역에 따라 경유 노드를 설정하고 이를 이용하여 경로 탐색을 수행하기 때문에 RRT-GRID 알고리즘이라고 명명하였다.

RRT-GRID 알고리즘의 경유 노드는 처음 3개의 임시 경유 노드를 설정 후 도달 가능 여부를 판단하여 2개 미만의 경유 노드에 도달 가능할 경우 추가로 4개의 임시 경유 노드를 설정하여 최대 5개의 경유 노드를 가질 수 있도록 한다. 본 논문에서는 경유 노드가 일정 수준을 초과할 경우 경로 탐색 시간이 빠르게 증가하는 문제를 극복하기 위해 최대 5개의 경유 노드를 가지도록 경유 노드의 수를 제한하였다.

처음 3개의 임시 경유 노드 중 2개 미만의 경유 노드에 도달 가능할 경우 추가로 4개의 임시 경유 노드를 설정하는 이유는 경유 노드의 수가 2개 미만일 경우에는 빠른 경로 탐색이 가능하지만 기존 RRT 알고리즘과 비교하여 경로 길이를 크게 단축할 수 없기 때문이다. 경로 길이를 단축하기 위하여 초기 임시 경유 노드 3개 중 2개 미만의 경유 노드만 도달이 가능할 경우 추가로 4개의 임시 경유 노드를 설정 후 도달 가능 여부를 판단한다.

Fig. 2는 RRT-GRID 알고리즘의 Phase 1을 나타낸 그림이다. Phase 1에서는 출발 노드 및 도착 노드의 좌표를 바탕으로 그리드를 나누어 총 3개의 임시 경유 노드를 그리드의 꼭짓점에 설정한다. 3개의 임시 경유 노드를 설정한 후 출발 노드에서 임시 경유 노드로의 도달 가능 여부를 판단한다. 출발 노드에서 임시 경유 노드로의 도달 가능 여부를 빠른 확인을 위하여 양방향에서 트리를 구축하는 RRT-Connect 알고리즘[14]을 사용하였다.

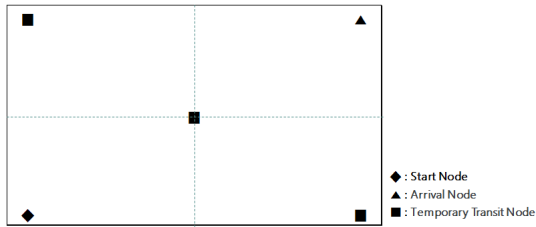


Fig. 2. Phase 1 of RRT-GRID Algorithm

RRT-Connect 알고리즘은 출발 노드와 도착 노드에서 각각의 트리를 하나씩 구축하여 총 2개의 트리를 활용하여 출발 지점과 목표 지점 사이의 경로를 찾아내고 출발 지점 트리와 목표 지점 트리가 만나면 두 트리를 연결하는 방식으로 경로를 탐색하는 알고리즘이다. RRT-Connect 알고리즘은 기존의 RRT 알고리즘보다 계산시간이 빠르고 출발 노드 및 도착 노드에서 트리를 구축하여 경유 노드의 도달 가능 여부를 확인하는 것에 적합하다[15].

3개의 임시 경유 노드에 대해 RRT-Connect 알고리즘을 통해 도달 가능 여부를 확인하고 도달 불가능한 노드들을 삭제하고 도달 가능한 임시 경유 노드는 경유 노드로 확정한다. Phase 1에서 도달 가능 경유 노드가 2개 이상 확정되었다면 임시 경유 노드 4개를 추가하는 Phase 2는 진행하지 않는다. 그러나 2개 이상의 노드에 도달 불가능하여 확정된 경유 노드가 한 개 이하일 경우 효율적인 경로 최적화를 위하여 임시 경유 노드 4개를 추가로 설정하는 Phase 2를 진행한다. 추가 임시 경유 노드 4개는 Fig. 3과 같이 설정한다.

Fig. 3은 Phase 1에서 도달 가능한 임시 경유 노드의 수가 2개 미만일 경우 추가로 4개의 임시 경유 노드를 설정하는 Phase 2를 나타낸 것이다. 추가된 임시 경유 노드들 또한 RRT-Connect 알고리즘을 통해 도달 가능 여부를 확인하여 도달 불가능한 임시 경유 노드는 삭제하고 도달 가능한 임시 경유 노드는 경유 노드로 확정한다. Phase 1에서 2개 이상의 경유 노드가 설정되었거나 Phase 2를 통해 추가 경유 노드 설정이 끝났다면 [출발

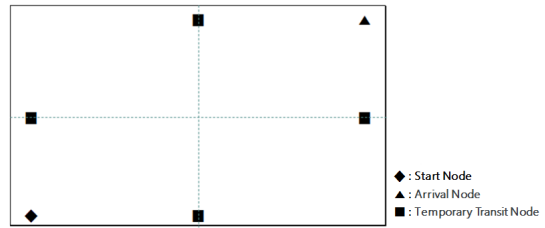


Fig. 3. Phase 2 of RRT-GRID Algorithm

노드 - 경유 노드 - 도착 노드] 사이의 경로 계획 최적화를 진행한다. 본 논문에서는 복잡한 구조 또는 좁은 영역에서 비효율적인 경로 계획 시간이 소요되거나 비효율적인 경로 길이가 도출되는 기존 RRT 알고리즘의 무작위 샘플링 방식을 대체하여 RRT-SMART(Shortest path Map Aware RRT) 알고리즘의 샘플링 방법을 사용하여 경로 탐색을 진행한다. RRT-SMART 알고리즘은 환경을 그리드 지도(Map)로 표현하고, 각 그리드 및 셀의 특성(예 : 장애물, 비용)을 저장하고 저장한 정보를 바탕으로 환경의 정보를 반영하여 장애물을 회피하는 방향으로 샘플링을 유도하는 방식이다. RRT-SMART 알고리즘은 장애물이 많고 영역이 좁은 환경에서 사용하기 적합한 알고리즘으로 지도 정보를 저장하여 장애물의 위치 또는 꼭짓점을 파악하여 장애물을 회피하는 샘플링을 통해 목적지까지의 경로를 탐색한다. RRT-SMART 알고리즘의 샘플링 방식을 사용하면 복잡한 장애물 및 좁은 영역에서 효율적인 경로 계획이 가능하다[6].

출발 노드에서 경유 노드까지 RRT-SMART 알고리즘을 통하여 최단 경로를 탐색하여 아크를 생성하고, 경유 노드와 다른 모든 경유 노드 또한 같은 방법을 통하여 최적 경로를 탐색한 후 아크를 생성한다. 다음으로 [경유 노드 - 도착 노드]를 RRT-SMART 알고리즘으로 경로를 탐색하여 아크를 생성한 후 마지막으로 [출발 노드 - 경유 노드 - 도착 노드]로 가장 비용이 낮은 최적 경로를 Dijkstra 알고리즘을 이용하여 도출한다.

본 논문에서 임시 경유 노드의 개수를 Phase 1에서 3개, 또는 Phase 2에서 추가로 4개를 설정하는 근본적인 이유는 2개 이상의 경유 노드가 존재할 경우 총 경로 길이의 단축에 효과적이기 때문이다. 즉, 경유 노드의 수가 2개 미만이라면 단순한 네트워크 모형으로 인해 계산 시간 단축은 가능하지만, 충분한 경유 노드가 존재하지 않아 경로 길이를 단축할 가능성이 감소하게 된다. 반대로, 경유 노드가 5개를 초과할 경우 경로 길이 단축 측면에서 유리하나 경로 탐색 시간이 지나치게 증가하는 문

제가 발생하기 때문에 최대 5개 이하의 경유 노드가 존재하도록 경유 노드의 수를 제한한다.

이상의 내용을 종합하여 본 논문에서 제안하는 RRT-GRID 알고리즘의 수행 절차를 정리하면 다음과 같다.

Step 1. (임시 경유 노드 3개 설정) 출발 노드 및 도착 노드를 기준으로 임시 경유 노드 3개 설정

Step 2. (RRT-Connect 기반 임시 경유 노드의 도달 가능 여부 탐색) 설정된 임시 경유 노드 3개에 대해 RRT-Connect 알고리즘으로 도달 가능 여부를 판단한 후 경유 노드 확정, 도달 불가 임시 경유 노드는 삭제

Step 3. 만약 경유 노드 2개 이상 확정시 Step 5로 이동, 그렇지 않을 경우 임시 경유 노드 4개 추가 생성 후 Step 4로 이동

Step 4. (RRT-Connect 기반 임시 경유 노드의 도달

가능 여부 탐색) 추가 설정된 4개의 임시 경유 노드의 도달 가능 여부 확인 후 도달 가능 경유 노드 확정하고 도달 불가능 임시 경유 노드 삭제

Step 5. RRT-SMART 알고리즘 기반 샘플링 통해 [출발 노드 - 경유 노드]에 대한 경로 탐색 및 아크 생성

Step 6. RRT-SMART 알고리즘 기반 샘플링 통해 [경유 노드 - 경유 노드]에 대한 경로 탐색 및 아크 생성

Step 7. RRT-SMART 알고리즘 기반 샘플링 통해 [경유 노드 - 도착 노드]에 대한 경로 탐색 및 아크 생성

Step 8. Dijkstra 알고리즘 통해 연결된 아크들에 대한 최적 경로 계산

Fig. 4와 Fig. 5는 RRT-GRID 알고리즘의 Phase 1 과 Phase 2 과정에 대한 예시를 나타낸 그림이다. RRT-GRID 알고리즘을 Phase 1과 2로 나누어 자세히

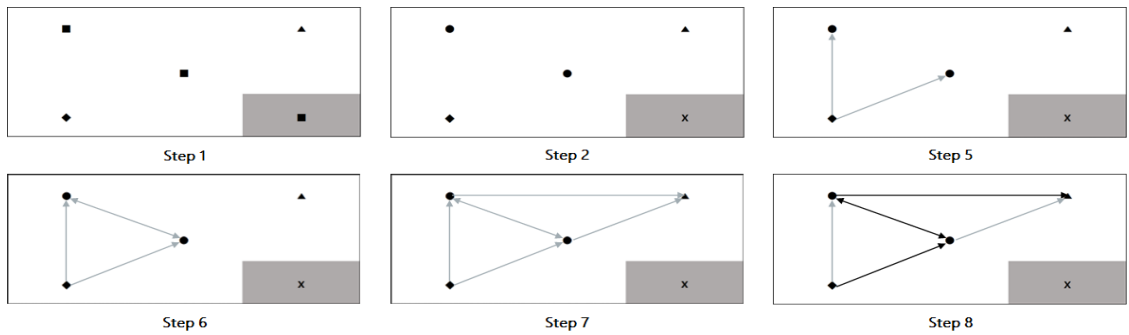


Fig. 4. Example of RRT-GRID Algorithm in Case of only Phase 1

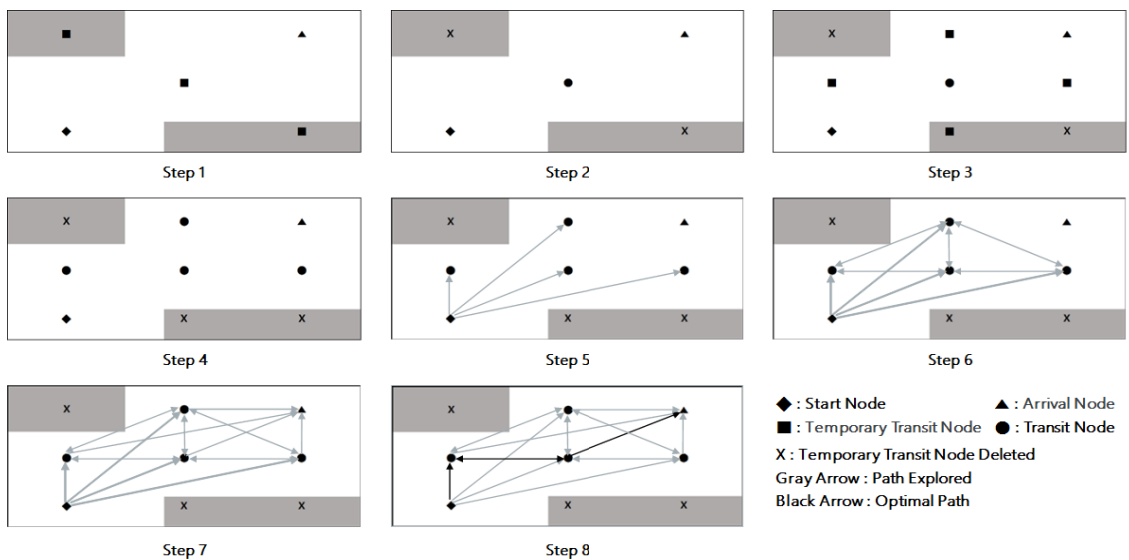


Fig. 5. Example of RRT-GRID Algorithm in Case of Phases 1 and 2

살펴보면 다음과 같다.

Fig. 4의 Phase 1에서는 Step 2의 3개의 경우 노드 가운데 도달 가능 경우 노드가 2개이므로 Step 4를 거치지 않고 Step 5를 진행하는 경우를 나타낸다. 즉 Step 2에서 임시 경우 노드의 도달 가능 여부를 RRT-Connect 알고리즘을 통하여 판별한 후 한 개의 도달 불가 임시 경우 노드를 삭제하고 2개의 경우 노드를 확정하였다. Fig. 4의 경우 확정된 경우 노드가 2개 이상이므로 Phase 2 과정은 생략되었다.

Fig. 5는 Step 2를 통해 확정된 경우 노드가 2개 미만이므로 Step 4를 실행하여 Phase 2를 진행하는 예를 보여 준다. Step 4에서는 추가로 설정된 4개의 임시 경우 노드의 도달 가능 여부를 RRT-Connect 알고리즘을 통하여 확인하여 도달 가능 경우 노드를 확정하고 도달할 수 없는 임시 경우 노드를 삭제한다.

Fig. 5의 Step 5에서는 출발 노드와 확정된 경우 노드를 RRT-SMART 알고리즘 기반 샘플링 방식을 통하여 연결하여 아크를 생성하는 과정을 나타낸다. Step 6에서는 확정된 모든 경우 노드들을 RRT-SMART 알고리즘 기반 샘플링 방식을 통하여 서로를 연결하여 아크를 생성한다. Step 7에서는 모든 [경유 노드 - 도착 노드]에 대해 RRT-SMART 알고리즘 기반 샘플링 방식을 통하여 연결하여 아크를 생성한다. Step 8에서는 연결된 아크들을 대상으로 Dijkstra 알고리즘을 통해 최적 경로를 도출한다.

3. 결과 분석

기존 RRT 알고리즘, RRT* 알고리즘과 본 연구에서 제안한 RRT-GRID 알고리즘의 성능 평가를 위해 실험 모형을 (1) 기본 모형, (2) 물류센터 1, (3) 물류센터 2, (4) 라인 형 생산시스템, (5) 셀 형 생산시스템 등 총 5개의 카테고리별로 나누고 카테고리별로 각 6개의 레이아웃을 구성하여 총 30개 모형에 대해 각각 100회의 시뮬레이션 수행 후 실험 결과의 평균을 비교 분석하였다.

실험 대상 모형은 실제 운용하고 있는 물류센터 및 제조 시스템의 환경을 최대한 반영하는 레이아웃 형태로 설계하였다. 실제 환경을 최대한 반영하여 실험 모형을 설정하였지만, AMR의 가감 속도, 회전속도 및 최대 속도, 로딩 및 언로딩 시간과 같은 실제 AMR의 물리적 요소는 반영하지 않았다.

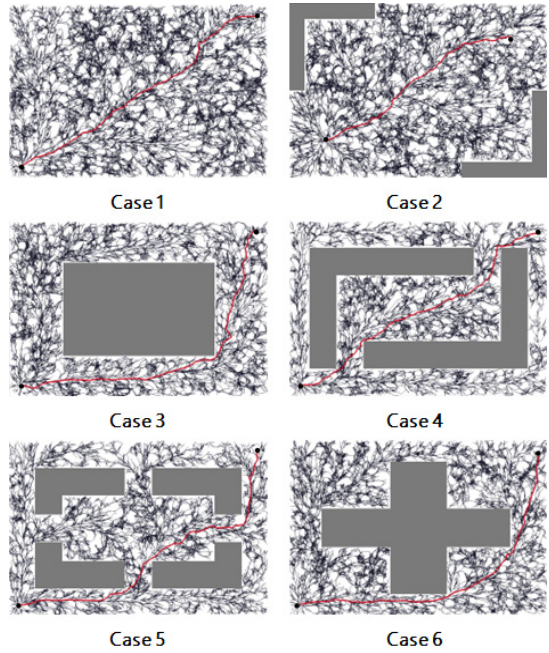


Fig. 6. Category 1: Layout of Basic Model

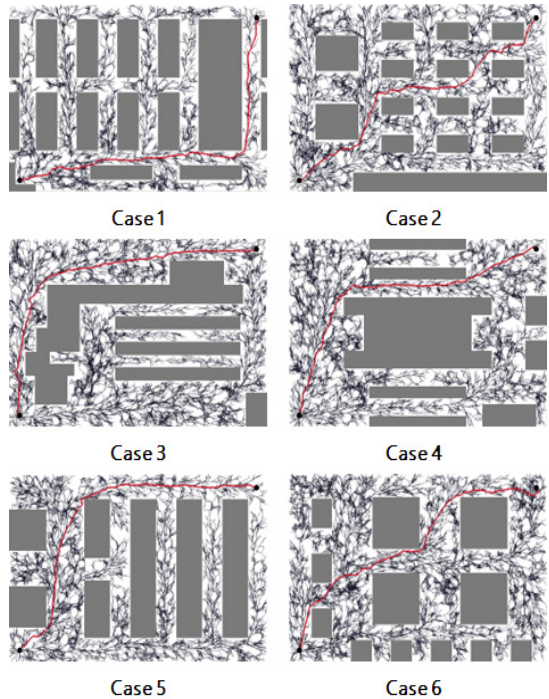


Fig. 7. Category 2: Layout of Distribution Center 1

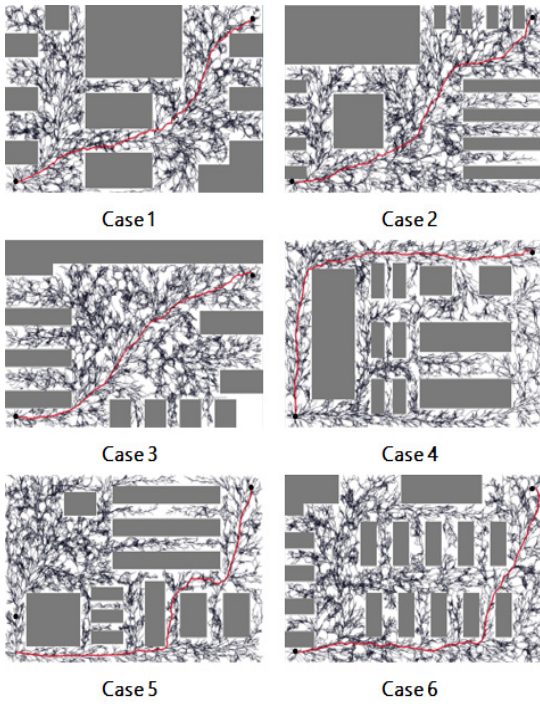


Fig. 8. Category 3: Layout of Distribution Center 2

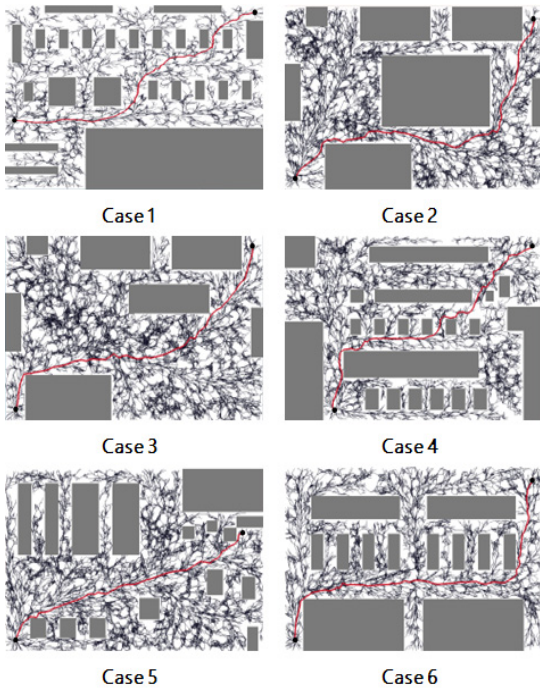


Fig. 9. Category 4: Layout of Cell-type Production System

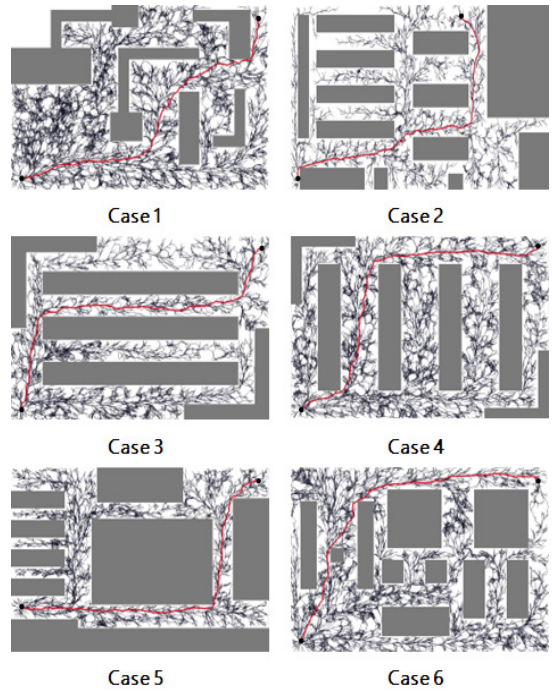


Fig. 10. Category 5: Layout of Line-type Production System

Fig. 6에서 Fig. 10은 카테고리 1에서 5를 구성하는 각 6개 모형의 레이아웃을 시각적으로 나타내고 있다. 출발 노드와 도착 노드가 장애물로 인해 도착 불가능한 일부 레이아웃을 제외하면 동일한 크기의 영역과 출발 노드 및 도착 노드를 설정하여 시뮬레이션을 진행하였다. 카테고리별 레이아웃의 상대적 복잡도는 실제 산업 환경을 고려하여 카테고리 1 < 카테고리 5 < 카테고리 4 < 카테고리 3 < 카테고리 2 순으로 나타낼 수 있도록 설계하였다.

Fig. 6은 기본 모형(카테고리 1)의 레이아웃을 나타낸 그림이다. 경로 계획 알고리즘의 성능을 테스트할 때 주로 사용하는 대표적인 모형이다. 장애물이 존재하지 않거나 그 수가 적은 환경으로 RRT 알고리즘과 같은 경로 계획 알고리즘이 장애물을 회피하고 목적지까지 탐색할 수 있는지를 확인할 때 사용되는 모형으로 복잡도가 가장 낮다[9,20].

Fig. 7은 물류센터 모형 1(카테고리 2)로서 풀필먼트 센터(Fulfillment Center)와 같은 B2C(Business-To-Consumer) 환경에 적합한 물류센터의 레이아웃을 표현하고 있다. 중앙에서 출하 장소까지 포장 또는 분류를 위한 넓은 라인이 존재하며, 맞춤형 수요(On-demand)와

높은 재고회전을 등으로 인하여 일반적으로 복잡도가 가장 높게 나타난다.

Fig. 8은 물류센터 모형 2(카테고리 3)로서 창고형 물류센터의 레이아웃을 표현하고 있다. B2B(Business-To-Business) 환경에 적합한 물류센터로 넓은 크기 및 다양한 공정을 포함하고 있다. 주로 재고를 보관하는 위치가 물류센터 외곽에 배치된 것이 특징으로 물류센터 중앙은 카테고리 2보다 넓으며, 이에 따라 많은 공정을 포함하고 있지만 넓은 중앙 공간을 보유하고 있고 상대적으로 장애물이 적고 재고회전율이 낮아 카테고리 2보다 복잡도가 낮다.

Fig. 9는 셀 형 생산시스템(카테고리 4)의 레이아웃을 나타내며, 공장 내 여러 개의 셀이 존재하는 것이 특징이다. 많은 셀의 존재로 인하여 실제 자율주행로봇을 운용하기에는 좁은 통로 등과 같은 복잡한 환경 및 다수의 자율주행로봇이 운용될 가능성을 가지고 있지만 셀 간의 거리가 자율주행로봇이 주행하기에는 충분하여 중간 정도의 복잡도를 나타낸다.

Fig. 10은 라인 형 생산시스템(카테고리 5)의 레이아웃을 나타낸다. 공장 내 여러 라인이 존재하는 형태다. 라인을 제외한다면 장애물이 존재하지 않으며 라인과 라인 사이도 자재를 공급하거나 공정이 완료된 반제품, 완제품을 운반하기 위하여 넓은 공간을 가지고 있는 것이 특징으로 라인과 라인 사이를 제외하면 자율주행로봇이 주행하기에 넓은 영역이 존재하고 라인이 있는 공간을 제외하면 장애물이 많이 존재하지 않으므로 기본 모형(카테고리 1)을 제외하고는 상대적으로 가장 넓은 영역 및 가장 낮은 복잡도를 나타낸다.

영역의 복잡도에 따라 본 논문에서 제안한 RRT-GRID 알고리즘이 얼마나 효율적인지를 알아보기 위하여 실제 환경을 반영한 시뮬레이션 모형을 설계하였고 RRT 알고리즘, RRT* 알고리즘, RRT-GRID 알고리즘을 Python Jupyter Notebook 6.4.5를 통해 구현하였다. 모든 카테고리 영역의 크기는 960×640픽셀로 동일하게 설정하였으며 시뮬레이션은 Windows 10, Intel(R) Core(TM) i7-7700 CPU @ 3.60GHz, NVIDIA GeForce GT 710, DDR4 8GB RAM 기반 컴퓨터를 활용하여 수행하였다.

Table 2는 실험 결과를 전체적으로 요약한 표이다. 모든 실험 환경에서의 결과를 종합하면 본 논문에서 제안한 RRT-GRID 알고리즘은 기존 RRT 알고리즘에 비해 경로 길이 약 8.61% 감소 및 경로 탐색 시간 약 31%의 개선된 결과를 나타냈다. 특히 장애물이 적은 환경보

Table 2. Summary of Simulation Results

Category	Algorithm	Measurement Factors	
		Path Length	Required Times (sec)
1	RRT	1183.75	260.96
	RRT*	1086.44	370.92
	RRT-GRID	1107.07	209.62
2	RRT	1350.66	1688.63
	RRT*	1250.15	2454.07
	RRT-GRID	1223.19	1031.80
3	RRT	1276.09	1767.04
	RRT*	1164.75	2340.52
	RRT-GRID	1163.42	1137.99
4	RRT	1251.07	1239.36
	RRT*	1140.11	1813.12
	RRT-GRID	1127.47	849.03
5	RRT	1214.69	573.33
	RRT*	1135.17	839.60
	RRT-GRID	1112.40	400.67

다는 장애물이 많고 복잡한 환경 또는 좁은 통로가 존재하는 환경에서 경로 탐색 시간을 크게 줄일 수 있었다.

기본 모형(카테고리 1)은 장애물이 없거나 간단한 장애물이 있는 모형으로 가장 낮은 복잡도를 나타낸다. RRT-GRID 알고리즘은 RRT 알고리즘 대비 평균 약 8.5%의 경로 길이 감소와 약 29%의 경로 탐색 시간 단축을 나타냈다.

본 연구에서 제안한 RRT-GRID 알고리즘은 카테고리 2와 3에 해당하는 물류센터 모형에 대해 가장 우수한 성과를 보임을 알 수 있다. 즉 본 연구의 취지와 부합하는 가장 복잡한 환경에서 가장 큰 경로 길이 감소와 경로 탐색 시간 단축 결과를 도출하였다. 현실적으로 물류센터 내 저장소와 포장, 제품 적재 장소와 연결된 라인이 있으면 AMR의 가동영역이 크게 줄어들기 때문에 이를 고려한 AMR의 운용이 필요할 것으로 생각된다. 물류센터에는 다수의 AMR이 사용될 가능성이 크며 작업자의 이동이 빈번하므로, 다수의 AMR과 작업자 사이의 충돌 예방을 고려한 경로 계획 및 AMR의 충전 장소를 고려한 운영 방안 등에 대한 추가적인 고려가 필요하다.

셀 형 생산시스템(카테고리 4)의 레이아웃은 넓은 공간에 여러 개의 셀을 배치한 것이 특징으로 다수의 셀이 AMR의 통행을 방해한다. RRT-GRID 알고리즘은 RRT 알고리즘 대비 경로 길이 약 10% 및 경로 탐색 시간 32%의 감축 효과를 각각 나타냈지만, 실제 환경에서는 다수의 AMR이 운용될 가능성과 작업자의 이동 가능성

이 존재하므로 작업 셀의 배치를 고려한 다수 AMR의 동선을 고려한 경로 계획이 필요할 것으로 생각된다.

라인 형 생산시스템(카테고리 5)은 라인과 라인 사이의 영역이 좁은 경우에는 AMR이 이동할 수 없는 환경이 존재하여 경우 노드를 통한 라인 전체를 회피하는 방식을 이용하여 경로 길이 약 8.5% 및 경로 탐색 시간 31%를 각각 단축할 수 있었다. 따라서 라인과 라인 사이에서 AMR을 운용하는 것보다는 복잡한 라인을 회피하여 AMR을 운용하는 전략이 바람직하다고 판단된다.

Fig. 11과 Fig. 12는 RRT 알고리즘의 경로 길이 및 경로 탐색 시간을 각각 100으로 설정하였을 때 RRT 알고리즘 대비 RRT* 알고리즘 및 RRT-GRID 알고리즘의 상대적인 결과를 보여 준다.

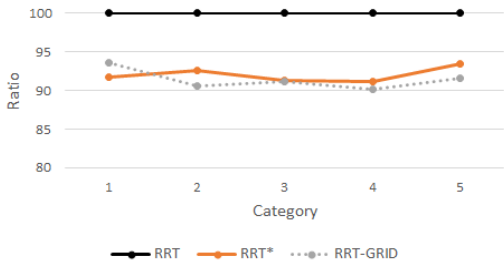


Fig. 11. Comparison Results of Path Length

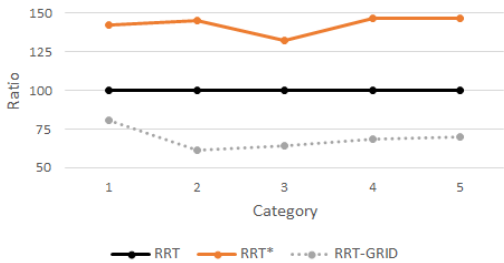


Fig. 12. Comparison Results of Required Times

RRT 알고리즘 대비 RRT-GRID 알고리즘의 경로 길이 단축 비율의 크기는 카테고리 1 < 카테고리 5 < 카테고리 3 < 카테고리 2 < 카테고리 4의 순으로 나타났다. 카테고리 4인 셀 형 생산시스템 모형을 제외하고는 카테고리별 복잡도에 비례하여 경로 길이를 감축하였다. 다른 모형에 비해 특히 카테고리 4에 대한 경로 길이 감축이 가장 큰 이유로는 셀 형 생산시스템 내에 많은 셀이 존재함으로 인하여 기존 RRT 알고리즘은 비효율적인 경로 탐색을 수행하였으나 RRT-GRID 알고리즘은 장애물을 회피하는 방식의 경로 탐색으로 장애물이 다수인 셀

을 제외한 공간을 효율적으로 탐색한 결과로 추측된다.

경로 탐색 소요 시간의 RRT 알고리즘 대비 RRT-GRID 알고리즘의 단축 비율은 카테고리 1 < 카테고리 5 < 카테고리 4 < 카테고리 3 < 카테고리 2의 순이며, 이는 카테고리별 복잡도와 동일한 패턴을 보였다. RRT-GRID 알고리즘은 장애물이 거의 없는 카테고리 1에 대해서는 기존 RRT 알고리즘 대비 약 80%의 경로 탐색 시간을 소요하였지만 가장 높은 복잡도를 가진 물류센터 모형 1 (카테고리 2)에 대해서는 약 60% 정도의 경로 탐색 시간만을 소요하였다.

4. 결론

본 논문에서는 최근 스마트공장과 스마트 물류 시스템 등에서 활용도가 증가하고 있는 자율주행로봇에 주로 탑재된 경로 계획 알고리즘인 RRT 알고리즘의 단점인 복잡하고 넓은 환경에서 경로 최적화 및 경로 탐색 시간의 취약성 문제를 개선한 RRT-GRID 알고리즘을 제안하였다.

다양한 실험을 통해 본 논문에서 제안한 RRT-GRID 알고리즘이 기존 RRT 알고리즘에 비해 전반적으로 우수한 결과를 나타냄을 확인하였다. 특히 RRT-GRID 알고리즘은 기존 RRT 알고리즘의 단점인 복잡하고 넓은 영역에서 경로 길이 및 경로 탐색 시간을 크게 단축하였으며 레이아웃의 복잡도가 높은 환경일수록 보다 효율적인 장애물 회피를 통해 상대적으로 더욱 우수한 결과를 도출하였다. 실제 환경에서 적절한 위치에 경우 노드를 설정하여 자율주행로봇을 운용한다면 보다 효율적으로 경로 탐색 시간 및 경로 길이를 감소시킬 수 있을 것으로 기대된다.

향후 연구과제로 보다 실제적인 동적 환경을 반영할 수 있는 알고리즘의 보완과 이에 대한 테스트가 필요하다. 실제 현장은 여러 작업자와 다수의 자율주행로봇이 공존하는 환경이 일반적이므로 작업장의 상황 변화에 대응하여 유연하고 빠르게 경로를 변경할 필요가 있기에 이를 반영할 수 있는 경로 최적화 연구가 이루어져야 할 것이다. 이 밖에도 본 논문의 시뮬레이션 환경에서 반영하지 못한 로봇의 물리적 요소인 물류 로봇의 주행속도, 회전속도, 충전 시간, 가동시간, 최대 부하량 및 연산 속도 등의 요소를 고려할 수 있어야 자율주행로봇을 보다 효율적으로 운용할 수 있을 것이다.

References

- [1] H. C. Hwang, S. H. Song, "A Study on the Factors Affecting the Acceptance of Logistics Robot in the Fulfillment Center Using the Technology Acceptance Model", *Journal of the Korea Academia-Industrial cooperation Society*, Vol.20, No.12, pp.287-297, 2019. DOI: <https://doi.org/10.5762/KAIS.2019.20.12.287>
- [2] J. E. Hwang, *A Study on the Path Planning and the Navigation Method for Collaboration of the Transportation Robot in the Smart Logistics Center*, Master's Thesis, Graduate School of Hankyong National University, 2019.
- [3] K. M. Kwak, B. Park, E. J. Ko, C. J. Yoon, K. J. Kim, "Rapidly Spreading Logistics Robot Applications", *Journal of Korea Robotics Society*, Vol.17, No.4, pp.387-396, 2022. DOI: <https://doi.org/10.7746/ikros.2022.17.4.387>
- [4] H. S. Kim, W. H. Kim, "Design and Implementation of a ROS-based Modular AMR", *Journal of Korean Institute of Information Technology*, Vol.20, No.5, pp.39-46, 2022. DOI: <https://doi.org/10.14801/ikjitt.2022.20.5.39>
- [5] S. Karaman, E. Frazzoli, "Sampling Based Algorithms for Optimal Motion Planning", *The International Journal of Robotics Research*, Vol.30, No.7, pp.846-894, 2011. DOI: <https://doi.org/10.1177/0278364911406761>
- [6] F. Islam, J. Nasir, U. Malik, Y. Ayaz, O. Hasan, "RRT*-SMART: Rapid Convergence Implementation of RRT* Towards Optimal Solution", *Proceedings of 2012 International Conference on IEEE, Mechatronics and Automation(ICMA)*, pp.1651-1656, 2012. DOI: <https://doi.org/10.1109/ICMA.2012.6284384>
- [7] H. B. Lee, H. K. Kwak, J. W. Kim, C. W. Lee, H. J. Kim, "Improvement of Online Motion Planning based on RRT by Modification of the Sampling Method", *Journal of Institute of Control, Robotics and Systems*, Vol.22, No.3, pp.192-198, 2016. DOI: <https://doi.org/10.5302/I.ICROS.2016.15.0135>
- [8] D. Connell, H. M. La, "Dynamic Path Planning and Replanning for Mobile Robots Using RRT*", *2017 IEEE International Conference on Systems, Man, and Cybernetics(SMC)*, 2017. DOI: <https://doi.org/10.1109/SMC.2017.8122814>
- [9] J. M. Park, *A Study on RRT-based Optimal Route Setting Method for Autonomous Mobile Robots for Logistics Work*, Master's Thesis, Graduate School of Korea Aerospace University, 2020.
- [10] R. Seif, M. A. Oskoei, "Mobile Robot Path Planning by RRT* in Dynamic Environments", *International Journal of Intelligent Systems and Applications*, Vol.2015, No.5, pp.24-30, 2015. DOI: <https://doi.org/10.5815/ijisa.2015.05.04>
- [11] M. Otte, E. Frazzoli, RRT X : Real-Time Motion Planning / Replanning for Environments with Unpredictable Obstacles Algorithmic Foundations of Robotics XI, Part of The Springer Tracts in Advanced Robotics Book Series (STAR, Vol.107), pp.461-478, 2015.
- [12] J. D. Gammell, S. S. Srinivasa, T. D. Barfoot, "Informed Sampling for Asymptotically Optimal Path Planning", *IEEE Transaction on Robotics*, Vol.34, No.4, pp.966-984, 2018. DOI: <https://doi.org/10.1109/TRO.2018.2830331>
- [13] H. T. Tak, C. G. Park, S. C. Lee, "Improvement of RRT*-Smart Algorithm for Optimal Path Planning and Application of the Algorithm in 2 & 3-Dimension Environment", *Journal of the Korean Society for Aviation and Aeronautics*, Vol.27, No.2, pp.1-8, 2019. DOI: <https://doi.org/10.12985/ksaa.2019.27.2.001>
- [14] J. G. Kang, D. W. Lim, Y. S. Choi, W. J. Jung, "Improved RRT-Connect Algorithm Based on Triangular Inequality for Robot Path Planning", *Sensors*, Vol.21, No.1, 333, 2021. DOI: <https://doi.org/10.3390/s21020333>
- [15] J. J. Kuffner, S. M. LaValle, RRT-connect: "An Efficient Approach to Single-query Path Planning", *In Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, CA, USA, pp.995-1001, 2000. DOI: <https://doi.org/10.1109/ROBOT.2000.844730>
- [16] A. Perez, R. Platt, G. D. Konidaris, L. P. Kaelbling, T. Lozano-Perez, "LQR-RRT* : Optimal Sampling-Based Motion Planning with Automatically Derived Extension Heuristics", *Proceedings of IEEE International Conference on Robotics and Automation*, pp.2537-2542, 2012. DOI: <https://doi.org/10.1109/ICRA.2012.6225177>
- [17] J. Li, S. Liu, B. Zhang, X. Zhao, "RRT-A* Motion Planning Algorithm for Non-holonomic Mobile Robot", *Proceedings of SICE Annual Conference 2014 September*, pp.1833-1838, 2014. DOI: <https://doi.org/10.1109/SICE.2014.6935304>
- [18] S. W. Hong, W. I. Son, K. H. Park, S. T. Kwun, I. S. Choi, S. W. Cho, "Design of Preprocessing Algorithm for HD-Map-based Global Path Generation", *The Journal of the Korea Institute of Intelligent Transport Systems*, Vol.21, No.1, pp.273-286, 2022. DOI: <https://doi.org/10.12815/kits.2022.21.1.273>
- [19] N. E. Park, J. Y. Mun, Y. A. Jeoung, S. Y. Cho, W. W. Huh, "Development of Halfway Station Recommendation Application Using Dijkstra's Algorithm", *Journal of Korea Multimedia Society*, Vol.24, No.2, pp.312-319, 2021. DOI: <https://doi.org/10.9717/kmms.2020.24.2.312>
- [20] H. B. Lee, "Path Optimization and Re-generation Algorithm for Indoor Mobile Robots in Obstacle Environments", *Journal of Institute of Control, Robotics and Systems*, Vol.25, No.7, pp.633-640, 2017. DOI: <https://doi.org/10.5302/I.ICROS.2019.19.0090>

김 성 현(Sung Hyun Kim)

[준회원]



- 2021년 2월 : 인제대학교 산업경영공학과 (공학사)
- 2023년 8월 : 인제대학교 일반대학원 AI융합로봇학과 (공학석사)

<관심분야>

물류 및 SCM, 생산계획 및 통제, 자율주행로봇

권 익 현(Ick-Hyun Kwon)

[정회원]



- 2000년 2월 : 고려대학교 산업공학과 (공학석사)
- 2006년 2월 : 고려대학교 산업공학과 (공학박사)
- 2007년 1월 ~ 2008년 1월 : University of Illinois at Urbana-Champaign, 박사후연구원
- 2008년 3월 ~ 현재 : 인제대학교 산업경영공학과 교수

<관심분야>

물류 및 SCM, 생산계획 및 통제, 산업보안