

협업 로봇을 위한 연속적 검증 환경 및 동적 돌발 대응 기법

Continuous Verification Environment and Dynamic Incident Response Techniques for Collaborative Robots

이재호¹·홍민지²·김희연²·김성민²·최한림[†]

Jae-Ho Lee¹, Min-Ji Hong², Hee-Yeon Kim², Seong-Min Kim², Han-Lim Choi[†]

Abstract: Recent advancements in unmanned robot technology have significantly enhanced individual robot performance, expanded their mission capabilities and drawing attention to collaborative potential. Previous research lacks sophisticated environments for handling interrupted missions and complex collaboration scenarios when unexpected tasks arise. Therefore, we propose an advanced mission allocation environment to fully utilize highly intelligent unmanned robots. We also propose a greedy algorithm-based mission allocation and replanning algorithm as a baseline performance metric in this environment. We conducted 1000 experiments for each combination of robot and task quantities to compare the global optimum solution with our proposed greedy algorithm. Our results demonstrate that when dealing with a large number of tasks, the proposed algorithm achieved approximately 90% reduction in computation time with 3.6% to 21% performance gap compared to the optimal solution.

Keywords: Multi-Robot System, Task Allocation, Unexpected Event, Collaborative Task

1. 서론

최근 무인 로봇 기술의 비약적인 발전으로 개별 무인 로봇의 임무 수행 범위가 크게 확장되고 있다. 과거에는 하드웨어의 계산 능력 한계로 인해 단일 로봇이 특정 임무만을 수행할 수 있었으나, 컴퓨팅 성능과 제어 기법의 발전으로 이제는 다양한 임무를 연속적으로 처리할 수 있게 되었다. 이러한 발전은 비용 효율성 측면에서도 큰 진전을 가져와, 동일한 비용으로 더 많은 로봇을 운용할 수 있게 되었다. 이에 따라 다수의 이기종 로봇들이 협업하여 복합 임무를 수행하는 군집 로봇 시스템에 대한 연구가 활발히 진행되고 있다^[1].

군집 로봇은 개미, 벌, 새 떼와 같은 생물학적 군집에서 영감을 받은 분산 제어 메커니즘을 활용한다. 크게 중앙 집중형 제어 방식과 자율 분산형 협업 방식으로 구분되는데, 개별 로봇들

의 단순한 상호작용만으로도 복잡한 집단 행동 패턴을 만들어 낼 수 있다. 특히 군집 로봇은 일부 로봇의 고장에도 전체 시스템 기능을 유지하는 강건성, 환경 변화에 유연하게 대응하는 적응력, 그리고 로봇의 추가와 제거가 용이한 확장성 등의 장점을 가진다. 본 연구에서는 이러한 군집 로봇 시스템 중에서도 중앙 제어 방식에 초점을 맞추어 연구를 진행하였다.

군집 로봇 연구는 크게 제어 방식, 개별 로봇 알고리즘, 운용 환경의 세 가지 분야로 구분할 수 있다. 특히 Han^[2]은 거대 언어 모델(LLM)을 기반으로 한 고도화된 군집 로봇 환경에서 필요한 연구 방향을 제시하였다. LLM의 특성상 각 로봇이 인간 수준의 지능을 보유할 것으로 가정하고, 이러한 고도의 지능을 가진 로봇들을 어떻게 효과적으로 제어할 수 있을지에 대해 연구하였다. 이는 기존 로봇의 특성을 포괄하면서도 한층 발전된 형태의 로봇을 상정한다는 점에서, 향후 군집 로봇 연구의 방향성을 잘 보여준다.

지능적인 군집 로봇을 연구하고 활용하기 위해서는 다양한 상황을 정교하게 고려할 수 있는 고도화된 환경이 필수적이다. 그러나 기존의 임무 할당 연구에서 사용된 환경들은 주로 임무 수행 순서만 결정하거나 타임스텝 기반의 임무 할당에 국한되어 있었다. 연속적인 환경에서 진행된 연구들도 대부분 임무간의 이동만 연속적으로 고려했을 뿐, 임무 자체를 연속적으로 다

Received : Jan. 24, 2025; Revised : Apr. 6, 2025; Accepted : Apr. 28, 2025

* This work was supported by the Agency For Defense Development grant funded by the Korean Government in 2025.

1. Principal Researcher, Aerospace Engineering, KAIST, Daejeon, Korea (16jhlee@kaist.ac.kr)

2. Manager, Aerospace Engineering, KAIST, Daejeon, Korea (alswl, dkssud715, bcn05002@kaist.ac.kr)

† Associate Professor, Corresponding author: Aerospace Engineering, KAIST, Daejeon, Korea (hanlimc@kaist.ac.kr)

문 연구는 상대적으로 부족한 실정이다^{3,4)}. 이러한 한계점은 예상치 못한 돌발상황에 대응할 때도 명확히 드러난다. 현재의 환경에서는 돌발임무를 기존 임무 전후에 단순히 할당하거나, 기존 임무를 포기하고 새 임무를 수행하는 등의 제한적인 방식만 가능하기 때문이다.

지능적인 군집 로봇은 로봇 간 협업을 통해 임무를 더욱 효율적으로 해결할 수 있는 장점을 지니고 있다. 이러한 장점을 극대화할 수 있는 임무 할당 연구를 위해서는 앞서 언급한 한계점을 보완하면서 개별 임무 수행 시 협업이 가능한 발전된 환경이 요구된다. 따라서 본 논문에서는 임무 할당 알고리즘을 제시하기에 앞서, 이러한 요구사항을 충족시키는 고도화된 임무 할당 환경을 제안한다.

우리가 제안하는 환경은 연속적인 시간축에서 임무 수행 및 임무 간 이동이 가능하며, 각 임무는 여러 로봇이 함께 협업하여 수행할 수 있도록 설계되었다. 이 환경에서는 협업을 통해 기존 임무를 더 빠르게 완료할 수 있으며, 각 임무의 특성에 맞게 협업 방식이 조정된다. 단순히 여러 대의 로봇을 할당하면 임무가 빨리 끝나는 것이 아니라, 각 로봇의 특성을 반영하여, 할당되는 로봇의 종류에 따라 협업 방식이 달라지도록 설계하였다.

또한 임무 추가 및 협업 방식 확장을 용이하게 하여, 추후 원하는 임무를 쉽게 생성하고 연구에 활용할 수 있도록 환경을 구축하였다. 단순 임무 할당을 넘어 돌발 임무 할당 연구가 가능하도록 임무 수행 도중 돌발 임무가 생성될 수 있게 하였다.

본 논문의 주된 기여점은 협업 임무를 수행하던 로봇이 돌발 임무에 대응하기 위해 이탈할 경우, 남은 로봇들이 해당 협업 임무를 단독으로 계속 수행할 수 있도록 하여 시간축에서의 연속적인 환경을 고려했다는 점이다. 또한 돌발 대응을 마치고 복귀한 로봇은 협업 및 단독 수행에 따른 진척도를 계산하여, 현재까지 진행된 진척도에서부터 효율적으로 다시 협업 임무에 참여할 수 있다.

이러한 접근은 연속적인 시간축에서 다양한 종류의 임무 중단 및 재개와 돌발 상황 대응을 모두 고려할 수 있는 환경을 제공함으로써, 고도화된 군집 로봇의 임무 할당 및 돌발 대응 연구에 실질적인 기여를 하였다.

군집 로봇의 주요 도전 과제로는 각 로봇의 특화된 기술을 최대한 활용하는 임무 할당 최적화, 로봇 간 반복적 인지정보 통신과 토론을 통한 강건한 추론 능력 구축, 복잡한 계층적 컨텍스트 정보 관리, 그리고 다양한 유형의 메모리 관리 체계 구축 등이 있다⁵⁾. 이 중에서도 임무 할당 최적화는 다른 연구 내용들을 포괄하는 핵심적인 과제로서, 본 연구에서는 각 로봇의 능력과 군집의 특성을 활용한 임무 할당 최적화 방안을 제시하고자 한다.

임무 할당 최적화 방식은 크게 규칙 기반 알고리즘, 휴리스틱 알고리즘, 강화학습 방식으로 나눌 수 있다. 그러나 규칙 기

반 알고리즘은 중앙 집중형 시스템에서만 적용 가능하며, 군집의 규모가 커질수록 적용이 어렵다는 한계를 가진다. 따라서 향후 다수의 이중 로봇으로 구성될 군집 로봇 환경에서는 규칙 기반 알고리즘의 활용이 제한적일 것으로 판단된다.

다수의 로봇으로 구성된 환경에서 전역 최적해를 도출하는 것은 상당한 계산 비용이 요구되므로, 실용적 관점에서는 휴리스틱 알고리즘을 통한 지역 최적해 도출이 선호된다. 대표적인 휴리스틱 알고리즘으로는 선형 최적화, 혼합 정수 선형 계획법(MILP)과 같은 수리 계획법 기반 알고리즘과 그리디(greedy) 알고리즘과 같은 탐색 기반 알고리즘이 있다. Jing⁶⁾의 연구에 따르면, 군집 로봇 환경에서 MILP를 이용한 임무 할당은 상당한 한계를 보여준다. MILP는 NP-hard 문제로서, 각 로봇의 특성이 상이하고 다양한 임무가 존재하는 연속적인 환경에서는 변수와 제약 조건이 기하급수적으로 증가하여 전통적인 수학적 방법으로는 해결이 어렵다.

이러한 한계를 극복하고자 Jing은 기계학습 기법과 MILP를 결합한 하이브리드 접근 방식을 제안하였다. 이 방식은 순수 MILP방식에 비해 향상된 성능을 보였으나, 이는 역설적으로 복잡한 수리 계획법 기반의 휴리스틱이 군집 로봇 문제 해결에 적합하지 않음을 시사한다. 따라서 본 연구에서는 수리 계획법 기반의 복잡한 휴리스틱이 아닌, 그리디 알고리즘과 같은 단순하면서도 효율적인 탐색 기반 휴리스틱을 통해 복잡한 군집 로봇 환경에서의 임무 할당 문제에 접근하고자 한다.

한편, Jingjing⁷⁾은 다중 로봇 강화학습을 활용하여 복잡한 임무 할당 문제를 해결하는 접근 방식을 제시하였다. UAV로 구성된 군집 로봇 환경에서, 각 UAV가 상호 간 정보 교환 없이 자율적으로 동작하며 임무 할당을 최적화하는 것을 목표로 하였다. 이를 위해 여러 로봇이 자율적으로 단일 임무를 수행할 수 있는 다중 로봇 강화학습 방식을 채택하였다. Q-러닝 기반의 다중 로봇 강화학습을 통해 중앙 집중형 방식에 근접한 성능을 달성했다는 점에서 주목할 만하다.

그러나 이러한 강화학습 기반 접근법은 순차적 임무 할당을 다루는 본 연구의 맥락과는 차이가 있다. 대부분의 강화학습을 사용하는 연구는 각 로봇이 완전히 자율적으로 움직이며 군집 로봇의 하위 단계 알고리즘인 개별 로봇 제어에 초점을 맞추고 있는 반면, 본 연구는 상위 단계의 임무 할당 최적화를 목표로 한다. 따라서 다중 로봇 강화학습은 본 연구의 직접적인 해결책이라기보다 제안된 상위 단계의 임무 할당 최적화 방식과 연계하여 사용할 수 있는 접근 방식으로 판단된다.

우리는 두 번째 기여점으로 새로운 임무 할당 및 재계획 알고리즘을 함께 제안한다. 복잡한 환경에서도 효율적인 임무 할당과 실시간 대응이 가능하도록, 탐색 기반 휴리스틱 알고리즘을 설계하였다. 시뮬레이션 실험을 통해 제안된 알고리즘이 다양한 상황에서 효율적인 임무 할당을 수행할 수 있었음을 확인

하였으며, 특히 돌발 상황에 대한 임무 재계획, 임무 중단 및 재개, 새로운 임무 추가 등의 기능이 정상적으로 작동함을 검증하였다.

2. 선행 연구

본 연구는 다중 로봇 임무 할당의 고도화된 연속적인 검증 환경과 임무 할당 및 재계획 알고리즘 개발을 목표로 한다. 이를 위해 다중 로봇 임무 할당 문제 해결을 위한 주요 알고리즘, 동적 환경에서의 재계획 방법에 대한 선행연구를 검토하였다.

다중 로봇 임무 할당 문제를 해결하기 위한 알고리즘 연구는 크게 중앙 집중식과 분산형으로 나뉜다. 중앙 집중식 할당 알고리즘은 중앙 제어 시스템에서 모든 로봇의 임무를 계획하고 조율하는 방식으로, 높은 최적화 성능을 보인다. 분산형 할당 알고리즘은 각 로봇이 독립적으로 의사결정을 내리거나 분산 통신을 통해 협력하는 방식이다. 이는 유연성과 확장성 측면에서 강점이 있지만 전역 최적화에는 다소 한계가 있다.

중앙 집중식 다중 로봇 임무 할당 방식의 한 예시로, Liu^[8]는 A* 알고리즘과 유전 알고리즘을 결합한 새로운 임무 할당 방식을 제안하였다. A* 알고리즘은 로봇의 이동 비용 계산과 경로 계획에 활용되었으며, 유전 알고리즘은 최적의 임무 할당을 도출하는 데 사용되었다. 성능 향상을 위해 그리디 알고리즘을 활용하여 초기 population을 생성하고, PMX crossover 연산과 다양한 돌연변이 연산자를 적용하였다. 또한, 엘리트 전략을 도입하여 최적해를 보장하였으며, 이를 통해 효율적이고 최적화된 임무 할당이 가능함을 입증하였다.

Braquet^[9]는 분산 경매 기반의 동적 임무 할당 알고리즘인 Greedy Coalition Auction Algorithm (GCAA)을 제안하였다. 이는 공간적으로 분산된 다중 로봇 시스템에서 동적 임무 할당 문제를 해결하는 데 중점을 둔다. 각 로봇이 최대 하나의 임무만 할당 받을 수 있는 상황을 다루며, 여러 로봇이 동일 임무에 할당될 수 있음을 고려하여 로봇 간 협업을 반영하였다. 중앙 통제 없이 로봇이 개별적으로 협상하는 분산 구조를 가지며, 로봇의 현재 상태와 임무 조건에 따라 동적으로 임무 할당을 업데이트한다는 특징이 있다.

Zhang^[10]은 이기종 로봇 팀의 최적 임무 할당을 위한 새로운 접근방식인 Stochastic Clustering Auction (SCA)을 제안하였다. NP-hard 문제인 임무 할당 문제를 해결하기 위해 마르코프 체인 검색 프로세스와 Simulated Annealing을 결합한 방식을 제안하였다. SCA는 전역 최적화와 함께 사용된 최초의 확률적 경매 방식으로, 임무 클러스터 간의 확률적 전송 및 교환을 기반으로 임무 할당을 수행한다. 또한, Annealing 파라미터 조정과 상향 이동 여부에 따라 중앙 집중식 및 분산식 방식을 모두 지원함으로써 다양한 환경에서의 유연한 활용이 가능하다.

다중 로봇 임무 할당에서 초기 계획의 중요성은 잘 알려져 있으나, 동적으로 변화하는 환경에서는 재계획의 역할이 더욱 부각된다. 로봇이 수행중인 임무는 다양한 외부 요인에 의해 방해받을 수 있으며, 특히 돌발 상황에서는 기존 계획의 수정이 불가피하다. 따라서 재계획은 임무의 연속성을 유지하고, 시스템의 강건성을 확보하기 위해 필수적인 요소로 간주된다.

Chen^[11]은 동적 환경에서의 다중 로봇 임무 할당을 위해 재할당과 재스케줄링을 결합한 새로운 접근 방식을 제안하였다. 이 접근법은 불확실한 시점에 새로운 임무가 추가되거나 환경이 변화하는 상황에서, 현재의 임무 할당과 스케줄을 지속적으로 적응시킴으로써 다중 로봇 시스템의 성능을 유지하는 것을 목표로 한다. 임무 할당의 균형을 맞추고 시스템의 전반적인 효율성을 향상시키기 위해, 로봇 간 레벨에서 재할당을 진행하며, 재할당과 환경 변화의 영향에 대응하여 각 로봇의 효율을 유지하기 위해, 로봇 내 레벨에서 재스케줄링을 진행한다. 특히, 기존 연구에서 독립적으로 다루었던 재할당과 재스케줄링을 결합한 최초의 시도로 동적 환경 대응에서의 높은 성능을 입증했다는 점에서 의의가 있다.

Dai^[12]는 다중 로봇 시스템의 동적 작업 할당 문제를 해결하기 위해 세 가지 접근법을 제안하고 이를 비교하였다. 경매 기반 접근법은 최소 이동 비용을 기준으로 임무를 할당하고, 탐색 임무의 실행 순서를 최적화하며, 파괴 임무에 대해서는 로봇의 공격 능력을 반영한 입찰 조건을 적용하였다. 공석 연쇄 접근법은 로봇당 하나의 임무만 할당하되, 탐색 임무는 비그리디적 전략으로 선택하고 파괴 임무는 그리디적 전략으로 선택한 후, 보다 효율적인 로봇에게 임무를 재할당한다. 심층 Q-learning 접근법은 로봇과 임무의 상태를 고려하여 전략적 의사결정에서 임무 할당을 결정하는 신경망을 학습시킨다. 이 연구는 탐사 및 파괴라는 특수한 도메인에서 로봇의 능력과 타겟의 특성을 고려한 동적 임무 할당 문제를 다루었으며, 세 가지 접근법의 장단점을 비교 분석했다는 점에서 의의가 있다.

3. 검증 환경

본 장에서는 제안하는 알고리즘의 성능을 평가하기 위한 고도화된 검증 환경의 구조와 주요 특징을 설명한다. 개발된 검증 환경은 UGV와 UAV를 모두 포함하는 이기종 로봇 시스템을 지원하며, 연속적 환경에서의 임무 할당 및 재계획 시나리오를 효과적으로 검증할 수 있도록 설계되었다.

우리가 제안하는 검증 환경은 이종 로봇 시스템의 교전, 감시, 정찰, 위험물 탐지 등 다양한 특성을 가진 임무들의 수행을 모사할 수 있다. 로봇의 개체 수와 세부 특성(속도, 공격력, 센서 범위 등), 그리고 임무의 수와 세부 요구사항(수행 시간, 필요 자원 등)을 파라미터로 유연하게 조정할 수 있어, 각기 다른

환경 조건에서의 알고리즘 성능을 종합적으로 평가할 수 있다. 특히 같은 임무라도 로봇의 특성에 따라 수행 방식이 달라지는 점을 반영하여, 임무 할당 시 로봇의 개별 특성이 고려되도록 설계되었다.

본 연구는 기존 연구들의 격자 기반 이산 환경과 달리, 연속적인 공간에서의 로봇 움직임과 임무 수행을 지원한다. 이산화된 환경에서는 로봇의 위치와 이동이 특정 격자점으로 제한되어 실제 운용 환경을 정확히 반영하기 어려웠으나, 연속 환경에서는 로봇이 자유롭게 이동할 수 있어 보다 현실적인 시나리오 구현이 가능하다. 특히 정찰, 감시, 교전과 같은 임무들의 경우, 연속적인 환경에서는 로봇의 정확한 위치에 따른 센서 범위와 임무 수행 효과를 세밀하게 계산할 수 있어 보다 정교한 성능 평가가 가능하다.

본 환경은 지형 장애물이 존재하는 연속 공간에서의 경로 계획 최적화를 지원한다. 각 로봇은 주어진 임무를 수행하기 위해 장애물을 회피하며 목표 지점으로 이동해야 하는데, 이때 지형의 특성이 로봇의 이동에 미치는 영향을 고려한다. UAV의 경우 지형 제약 없이 일정한 속도로 이동할 수 있지만, UGV는 험지나 경사와 같은 지형 조건에 따라 이동 속도가 크게 달라진다. 이러한 특성을 반영하여 각 로봇의 실제 이동 시간을 정확히 예측하고, 이를 바탕으로 보다 현실적인 임무 할당이 이루어지도록 하였다.

감시 임무는 사전에 식별된 적의 위치에서 정보를 수집하는 것을 목표로 하며, 시간 제약 조건이 부여된 Time Window 기반으로 운용된다. 각 감시 임무는 고유의 수행 시간대가 지정되어 있으며, 이 시간 내에 임무를 완수해야만 높은 위험도를 지닌 적에 대해 효과적으로 대응할 수 있다. UAV의 경우 신속한 기동이 가능하여 Time Window 제약 조건을 충족시키기에 유리하나, UGV는 지형 조건에 따른 이동 시간 변동을 고려해야 한다. 각 감시 지점당 최대한 대의 로봇만이 배치되며, 이는 제한된 자원의 최적 활용을 가능하게 한다. 또한, 감시 임무는 특성상 연속성이 매우 중요하다. 따라서 로봇이 감시 임무 수행 중 다른 임무로 전환하거나 임무를 중단할 경우, 이는 감시 대상을 놓친 것으로 간주되어 즉시 임무 실패로 처리된다. 중단된 감시 임무는 대상에 대한 연속적인 정보 수집이 불가능해진다는 점에서 재개를 허용하지 않는다.

정찰 임무는 특정 영역 내 적의 탐지 및 위치 확인을 목표로 하며, 로봇의 이동 특성 및 센서 범위를 고려한 최적 경로 설계를 기반으로 수행된다. 로봇의 종류에 따라 임무 수행 방식은 상이하며, 특히 지형 조건이 각 로봇의 정찰 효율성에 미치는 영향은 차별적이다. UAV의 경우 지형 제약 없이 일정한 속도로 이동하며 넓은 시야각을 확보할 수 있어, 광역 정찰에 특화되어 있다. 높은 고도에서의 조망이 가능하여 험지나 접근이 제한된 지역까지 효과적으로 정찰할 수 있다는 장점이 있다. 반면

UAV는 지형 조건에 따라 이동 속도가 크게 변화하며, 특히 험지에서는 이동 속도가 현저히 감소하여 정찰 시간이 증가한다. 이는 전체 지도에 각 위치의 지형 정보가 표현되고 각 이동 및 임무를 계산할 때 이 지도의 값을 참조하여 수행하는 로봇의 속도를 조절한다. 이외에도 정찰 임무의 특성상 중단과 재개가 유연하게 이루어질 수 있다. 임무가 중단된 경우, 미 탐색 구역을 기준으로 새로운 경로점이 자동으로 재계산되어 효율적인 임무 재개가 가능하다. 특히 임무 재개 시 투입되는 로봇의 수가 변경되더라도, 남은 구역에 대한 최적 경로가 재설정되어 한 대의 로봇으로도 전체 영역을 정찰할 수 있도록 설계되었다.

위험물 탐지 임무는 잠재적 위험물이 설치될 것으로 예상되는 특정 구역을 대상으로 한다. 본 연구의 검증 환경에서는 위험물 탐지 센서와 같은 특수 장비를 갖춘 로봇만이 이 임무를 수행할 수 있도록 제한하여, 실제 운용 환경의 제약 조건을 반영하였다. 위험물은 다른 로봇들의 이동에도 영향을 미친다. 해당 구역은 잠재적 위험 지역으로 간주되어 UGV의 접근이 제한되며, 이에 따라 우회 경로를 이용하여야 한다. 따라서 위험물 탐지 임무의 신속한 완료는 전체 시스템의 효율성 향상으로 이어진다. 특히 해당 구역이 안전 지역으로 확보될 경우, 다른 로봇들의 이동 경로가 최적화되어 전체 임무 수행 시간이 단축된다. 위험물 탐지 임무의 중단과 재개는 정찰 임무와 동일한 메커니즘을 따른다. 임무 중단 시 미 탐지 영역을 기준으로 새로운 경로가 계산되며, 재개 시에는 투입 가능한 로봇의 수와 특성을 고려하여 잔여 구역에 대한 탐지 경로가 재설정된다. 단, 임무 재개 시에도 위험물 탐지 장비를 갖춘 로봇만이 투입될 수 있다는 제약은 유지된다.

교전 임무는 적을 섬멸하면서 아군의 피해를 최소화하는 것을 목표로 한다. 이 과정에서 아군 로봇의 생존율을 높이고, 손실되는 개체 수를 최소화하는 것이 핵심이다. 이는 연속적으로 이어지는 다양한 임무의 성공적인 수행을 위해 필수적인 요소이다. 현실에서의 교전 임무는 중단 및 재개가 불가능하며, 따라서 전투 중이던 로봇은 돌발 상황에 대응하기 위해 진행 중이던 전투를 중단하는 일은 발생하지 않는 것으로 가정한다. 임무의 진행 상태는 전투 중 발생하는 로봇의 상태 변화(예: 파손, 생존 여부)와 적 개체의 상태를 기반으로 실시간 계산된다. 각 로봇의 특성과 전투 능력(예: 이동 속도, 무장 성능, 방어력)에 따라 수행 차이가 발생할 수 있으므로, 특정 임무에 적합한 로봇의 배치를 최적화하는 알고리즘이 필요하다.

본 연구는 실증 환경 운용 시 발생할 수 있는 다양한 돌발 상황을 고려하여, 이에 대한 재계획을 포함한 검증 환경을 구축했다. 예를 들어, 이동 중 또는 정찰 중에 적을 발견했을 때 로봇은 해당 적을 감시하거나 교전하여 섬멸해야 하는 상황이 발생할 수 있다. 또한 이동 중이나 다른 임무를 수행하던 중 새로운 위험물이 추가로 발견되면, 해당 위험물을 제거해야 하며 이는 해

당 능력을 보유한 UGV만이 수행할 수 있는 제약이 따른다. 전투 중 파괴되거나, 이동 중 미 발견된 지뢰로 인한 고장 발생 등의 로봇의 기능적 문제가 발생할 수도 있다. 이 경우 해당 로봇의 후속 과업에 대해 대체 자원을 투입할 것인지 여부를 신속히 결정해야 한다. 다양한 돌발 상황을 반영함으로써, 알고리즘이 예기치 않은 상황에서 얼마나 유연하고 효과적으로 대응할 수 있는지를 평가할 수 있다. 이는 실제 운용 환경에서의 신뢰성을 높이는 데 기여한다.

4. 임무 할당 그리디 알고리즘

임무 할당에 있어 가장 중요한 것은 전체 임무를 끝내는 시간이 작아지는 것이다. 따라서 우리의 그리디 알고리즘은 두 단계의 구조를 가지며 첫 번째 단계에서는 임무의 효율적 할당을 최우선 목표로 하고 두 번째 단계에서는 전체 임무를 끝내는 시간이 줄어들도록 효율적인 임무 협업을 목표로 한다. 제안한 알고리즘이 두 단계로 나뉜 이유는 두 번째 단계만 있을 경우에는 임무 전체를 완료하지 않고 할당을 안 하는 경우가 생기기에 이는 임무 할당의 기본적인 목적을 만족하지 않는다고 생각하였다. 또한 단순히 임무를 개별 로봇만 수행하는 것이 아닌 여러 대가 할당되었을 때 좀 더 효율을 내는 환경을 설계하였기에 이를 그리디 알고리즘으로 해결하기에는 좀 더 정교한 설계가 필요하였다. 다음은 우리가 제안한 임무 할당 알고리즘이다 [Algorithm 1].

임무 할당 그리디 알고리즘의 2번째 줄인 각 임무에 대한 각 로봇 별 비용 계산은 그 로봇이 임무를 수행하였을 때 소모되는 자원(cost), 소모되는 시간(time), 얻는 보상(reward)의 가중합으로 계산된다. 또한 순차적으로 돌아가는 알고리즘이므로 이미 그 임무에 다른 로봇이 할당되어 있을 수 있다. 이 경우에는 이미 할당된 다른 로봇과 협업하여 임무를 수행하는 것을 상정하고 가중합을 계산한다. 임무를 협업하여 수행하는 방식은 임무 별, 로봇별로 다 다르고 우리가 제안한 환경에서는 이를 다 고려하여 가중합을 자동으로 계산해주는 함수를 사용한다.

[Algorithm 1] Greedy algorithm for task allocation

Algorithm 1. Greedy algorithm for task allocation

```

While True
  Calculate robot method cost for each task
  If all tasks are not yet completed:
    Assign the task with the lowest cost to a robot
  Elif assigning additional robot reduces total task duration:
    Assign the robot with the lowest cost to the task
  Else
    End algorithm

```

현재 할당된 로봇으로 모든 임무를 완료하지 못할 시 아직 임무 할당의 기본 조건인 전체 임무 완료를 만족하지 않았으므로 무조건적으로 가장 낮은 비용을 가지는 로봇을 할당하게 된다. 모든 임무가 완료되었을 경우 그때부터 두 번째 단계인 로봇 추가할당시 전체 임무시간이 줄어드는지 여부를 고려한다. 전체 임무시간이 줄어들게 된다면 추가적으로 가장 최적의 로봇을 한 대씩 할당하고 추가할당이 비효율적이어서 전체 임무시간이 늘어나게 된다면 더 이상 할당하지 않고 알고리즘을 종료한다.

5. 돌발 대응 그리디 알고리즘

돌발 상황에서 가장 중요한 것은 기존에 하던 임무를 중단하고 돌발 임무 대응 여부를 결정해야 되기에 임무의 특성에 따라 임무 중단과 재개를 어떻게 설계하는지가 가장 중요하다. 따라서 제안한 환경은 임무 수행을 위해 이동중, 임무 수행중의 경우에 임무 중단 및 재개를 고려할 수 있도록 설계되었고 돌발 발생 시각에 관계없이 항상 임무 중단 및 재개를 계산할 수 있다.

돌발 대응 그리디 알고리즘에 있어서도 우리는 돌발 대응 후 전체 임무 완료 시간을 최대한 적게 늘리는 것을 최우선 목표로 잡고 우리의 알고리즘을 설계하였다. 또한 교전 임무의 경우 그리디 알고리즘 특성상 교전에 승리하는 경우와 회피하는 경우에 대한 비교가 정확하게 일어나지 않을 수 있다. 우리는 이러한 문제를 방지하기 위해 돌발 상황이 교전 임무일 경우 실제 교전을 시행하기 전 사전 시뮬레이터를 통해 현재 할당된 로봇으로 교전을 시행하였을 경우 승패를 우선 파악한다. 사전 시뮬레이션 하에서 교전에 승리하기까지 순차적으로 로봇을 할당하고 승리하는 시점에서 추가 로봇을 할당할지, 교전 임무를 수행하지 않고 회피할지 결정한다. 다음은 우리가 제안한 돌발 대응 그리디 알고리즘이다 [Algorithm 2].

[Algorithm 2] Greedy algorithm for accident corresponding

Algorithm 2. Greedy algorithm for accident corresponding

For all accident corresponding method:

While True

Calculate robot method cost for accident task

If accident task is not yet completed:

Assign the robot with the lowest cost to an accident task

Elif assigning additional robot reduces total task duration:

Assign the robot with the lowest cost to the accident task

Else:

End while

Select corresponding method according to weighted sum

돌발 대응 그리디 알고리즘은 돌발 상황 인지 후 그 상황을 어떻게 대응할지 정해야 하기에 조건문을 하나 더 추가하였다. 최우선적으로 판단하는 것은 돌발 임무의 완료 유무이고 그 후 전체 임무시간 고려를 한다. 따라서 이 두가지 조건문에 의해 돌발 대응을 가장 잘할 수 있는 로봇의 조합을 찾게 된다. 그 후 마지막 조건문을 통하여 다른 돌발 대응 방식과 비교하여 가중합을 기준으로 선택한다.

돌발 대응 방식을 비교하기 위해서 우리는 세 가지 요소를 참고하였다. 첫번째로 전체 임무 달성률인데 이는 전체 임무를 얼마나 잘 수행하였는지 나타내고 각 임무별로 성공시에 얻을 수 있는 보상을 다르게 설계하였다. 예를 들어 교전에서 이겼을 시 탐색 임무를 성공하였을 때보다 더 많은 보상을 받게 된다. 성공시 보상 이외에도 각 임무의 달성률도 이 전체 임무 달성률에 고려된다.

두 번째로 고려하는 것은 전체 임무 수행완료 시간이다. 이 수행완료 시간의 가중치는 음수로서 임무수행시간이 줄어들수록 더 효율적인 임무 수행을 나타낸다. 마지막으로 고려하는 것은 임무 수행 시의 비용이다. 교전 임무의 경우 적 피해도 중요하지만 아군의 피해량도 중요하다. 따라서 이러한 항을 넣음으로서 비용까지 고려한 효율적인 알고리즘이 될 수 있도록 설계하였다. 가중합의 가중치는 이 알고리즘이 사용될 수 있는 환경과 임무가 변화될 수 있음을 고려하여 사용자가 조절할 수 있도록 설계하였다. 마지막 조건문이 들어간 이유는 돌발 상황을 완벽하게 대처하는 방식과 돌발 대응을 하지 않고 기존의 임무가 영향을 받지만 계속 수행하는 방식을 고려할 수 있는 환경이기 때문이다. 따라서 우리의 알고리즘은 다양한 환경 하에서 다양한 임무 방식을 고려할 수 있도록 설계하였다.

6. 실험 결과 및 결론

제한한 환경에서 제안한 그리디 알고리즘이 얼마나 잘 돌아가는지 알기 위해 완벽한 무작위 상황에서 임무 할당 및 돌발 대

응을 하도록 실험을 설계하였다. 초기 임무 할당 이후 전체 임무 완료 시간 사이에 무작위적으로 돌발 상황을 추가하고 이에 대해 돌발 대응을 효율적으로 하는지 확인하였다. 우리는 1000개의 상황에 대해서 초기 임무 계획 및 돌발 대응 알고리즘을 실험해보았고 문제없이 돌아가는 것을 확인할 수 있었다. 또한 우리가 제안한 그리디 알고리즘의 성능 비교를 위해 로봇 수와 임무 수별로 전역 최적해와 그리디 알고리즘의 성능 차이를 계산해보았다. 1000개의 상황에 대한 결과는 [Table 1]에 정리하였다.

[Table 1]은 각 알고리즘별 1000개의 시나리오에 대해서 돌발대응을 한 결과를 나타낸다. Reward는 돌발대응을 한 후의 전체 임무에 대한 보상으로서 높을 수록 좋은 값을 가진다. 이는 위에서 설명한 가중합과 같은 값으로 우리가 설정한 임무에 대하여 각각 보상, 수행시간, 비용을 지정해주었고 지정해준 로직대로 계산된 보상이다. Time은 전체 임무 수행 완료 시간이고 accuracy는 전역 최적해와 같은 돌발 대응 방식을 결정할 시나리오의 비율이다. Global은 전역 최적해(단순 전체 탐색) greedy는 우리가 제안한 그리디 알고리즘을 나타내며 임무는 기본적으로 4개의 정찰임무, 3개의 지뢰제거 임무, 2개의 교전 임무, 2개의 감시 임무로 구성되어 있으며 이의 2배수, 3배수, 4배수에 대해서 전역 최적해와 그리디 알고리즘의 성능을 비교하였다.

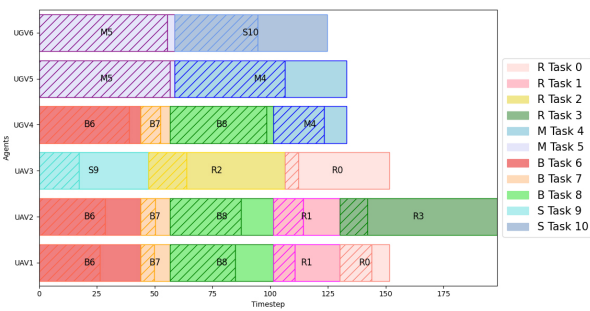
우리가 제안한 알고리즘인 그리디 알고리즘이 전체 임무와 로봇 수가 늘어날 수록 전역해를 구하는 시간에 비해 급격하게 계산시간이 줄어들 것을 [Table 1]을 통해 알 수 있다. 가장 적은 수의 임무 개수인 11개에서는 전역 최적해를 도출해내는데 걸리는 시간 보다 42.886% 감소하였고 가장 많은 수의 임무 개수인 44개에서는 약 90% 감소한 것을 볼 수 있었다. 또한, 전역 최적해 도출 과정에서는 개별 시나리오별 수행 시간에 상당한 편차가 존재하는 반면, 그리디 알고리즘은 시나리오와 관계없이 비교적 일관된 연산 시간을 나타내는 것으로 확인되었다. 보상의 값은 전역 최적값에 비해 적은 수에서는 44.411% 차이가 났지만 임무 수가 많아질수록 이 차이는 줄어들어 가는 경향을 나타

[Table 1] Experimental results

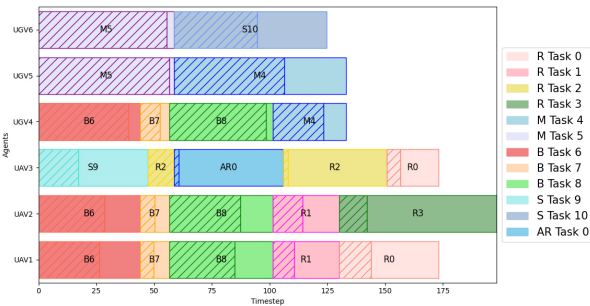
Scenario type	Global			Greedy			Global - Greedy difference by a scenario			
	Mean time(s)	S.T.D. time	Mean reward	Mean time(s)	S.T.D. time	Mean reward	Mean time(s)	Decrease time (%)	Mean reward	Performance gap (%)
3 uav / 3 ugv 11 missions	20.602	30.213	-88.044	11.766	10.855	-127.146	8.791	42.886%	38.906	44.411
6 uav / 6 ugv 22 missions	153.132	268.965	9.087	37.589	42.902	6.123	115.544	75.453	115.544	32.614
9 uav / 9 ugv 33 missions	485.366	833.652	96.567	71.447	84.001	93.098	412.3312	85.280	3.465	3.593
12uav/ 12ugv 44 missions	1212.967	2021.281	181.9256	121.3938	146.227	143.631	1086.266	89.992	38.325	21.050

념을 실험을 통해 확인할 수 있었다. 이는 복잡한 환경에서의 적용을 목표로 하는 그리디 알고리즘 개발 방향과 일치하며, 해당 맥락에서 의미 있는 결과로 평가된다.

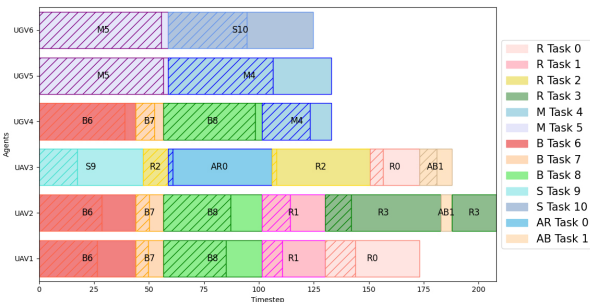
전체 1000개 시나리오에 대한 결과에 이어 각 시나리오별 돌발 대응 결과를 살펴보았다. 결과에 나오는 임무 이름은 약자로 표현되었다. R은 탐색 임무 그리고 M은 지뢰 제거 임무, B는 교전 임무, S는 감시 임무로 표현하였다. 또한 돌발 임무의 경우 각 임무의 약자 앞에 A를 붙여 표현하였다. 각 임무는 위치와 수행 시간이 다 다르고 UGV와 UAV도 종류별, 개체별로 다른 특성을 가지게 무작위성을 부여하였다. 아래의 [Fig. 1]~[Fig. 3]은



[Fig. 1] Initial Task allocation: The result of initial task allocation with 6 agents (3 UGV, 3 UAV) and 11 missions. Since R mission is search mission, it is allocated to UAV not UGV



[Fig. 2] Accident corresponding 1: The result of the accident corresponding after initial task allocation. AR Task 0 is an accident mission



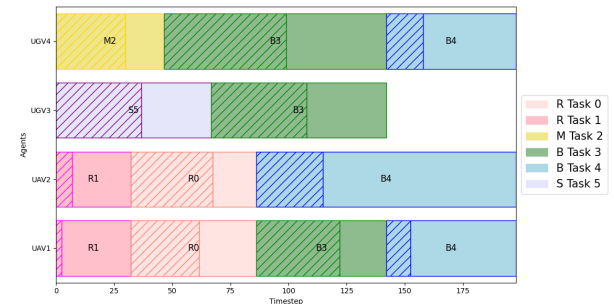
[Fig. 3] Accident corresponding 2: The result of accident corresponding after accident corresponding 3. AB Task 1 is accident mission 2

초기 임무 계획 및 돌발 대응의 한 결과이다.

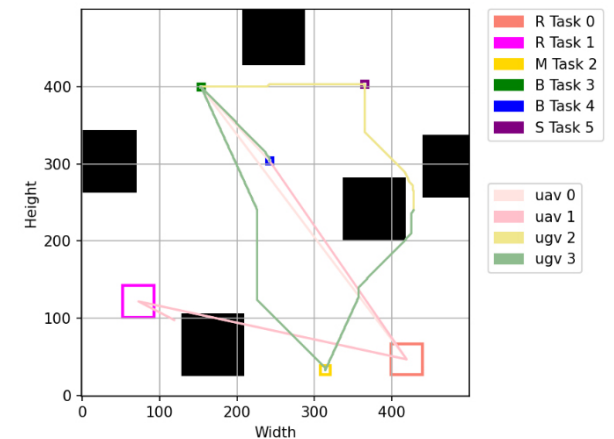
초기 할당을 할 때 R3이 비효율적으로 보인 것은 R 임무가 정찰 임무이기에 UGV가 하는 것보다 UAV가 하는 것이 더 효율적이어서 UAV가 하는 것으로 선택되었다. 그리디 알고리즘의 구조적 결점 역시 이 임무 할당 결과에 나오는데 마지막으로 할당되는 임무의 길이가 얼마나 길지 모르기에 다른 임무가 끝나도 R3을 계속 수행하도록 그리디 알고리즘이 할당하였다. 마지막에 할당된 R3 이외의 임무들은 효율적으로 비슷한 시간대에 끝날 수 있도록 할당이 되었고 임무 시간대가 다르게 시작하지만 같은 시간대에 협업도 하는 것을 볼 수 있다.

돌발 대응을 한 결과 [Fig. 2]를 보면 R2 임무를 하다가 돌발 대응을 하고 난 후에 다시 수행을 하는 것을 볼 수 있다. 두 번째 돌발 대응을 보면 UAV2와 UAV3이 협업하고 UAV2는 돌발 임무 수행 이후 중단했던 R3 임무를 다시 수행하는 것을 볼 수 있다. 이는 임무 중단 및 재개 방식이 잘 동작하고 있다는 것을 반증한다.

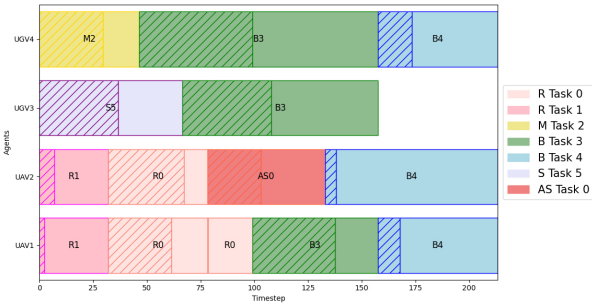
알고리즘이 제대로 돌아갔는지 검증하기 위해 적은 수의 로봇과 임무 환경에서 알고리즘을 돌려보고 그것을 2차원 지도에 그려보았다. 이는 [Fig. 4]~[Fig. 7]에 나타난다. 로봇의 시작



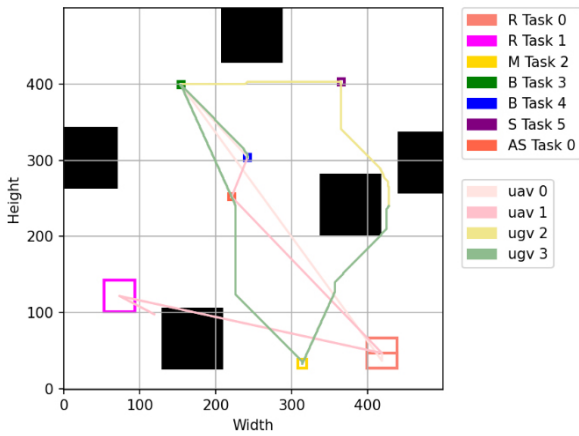
[Fig. 4] Initial Task allocation: The result of initial task allocation with 4 agents (2 UGV, 2 UAV) and 5 missions. We reduce the number of agents and missions in order to validate in 2D map



[Fig. 5] 2D map of Initial Task allocation: The result of initial task allocation with 4 agents (2 UGV, 2 UAV) and 5 missions



[Fig. 6] Accident corresponding 1: The result of the accident corresponding after initial task allocation. AM Task 0 is an accident mission



[Fig. 7] 2D map of Accident corresponding 1: The result of the accident corresponding after initial task allocation. AM Task 0 is an accident mission

지점은 [0, 0] 이고 UGV 2대와 UAV 2대 로 임무 할당을 하였다. [Fig. 6], [Fig. 7]의 경우는 돌발 대응을 한 결과로 2차원 지도의 가운데 지점에 돌발 임무가 발생하였다.

검은 지형은 UAV 만 지나갈 수 있고 UGV는 지나가기 힘든 험한 지형을 표현한 것으로 이 실험환경에서 UGV는 기존 속도의 15% 속도로 진행하는 것으로 설정을 하였다. 따라서 [Fig. 5]의 우측부분을 보면 UGV가 험난한 지형을 돌아가는 것을 볼 수 있다. 또한 UAV는 이 험난한 지형을 그대로 통과해서 지나간다. 본 환경은 이러한 험난한 지형을 지나갈 때 UGV가 어느 정도의 속도로 지나갈 지 설정을 할 수 있고 A* 알고리즘도 이를 반영하여 경로계획을 한다. 검은 부분 이외의 부분은 이동에 아무런 제약이 없는 평지 지역으로 설정을 하였다. 이러한 모든 정보는 지형 정보를 담아두는 변수에 초기 입력값으로 원하는 지형에 대한 정보를 주게 되면 원하는 환경에서 실험을 할 수 있다.

우리가 제안한 복잡하고 정교한 환경에서 임무 할당 및 돌발 대응 그리디 알고리즘이 잘 돌아가는 것을 확인할 수 있었다. 이에 대한 후속 연구로 우리가 제안한 복잡한 환경에서 그리디 알고리즘보다 빠르고 정교하게 돌아가는 알고리즘의 개발과

현실적인 임무 수행 방식을 본 환경에 추가하는 등의 방향이 존재한다.

7. 향후 연구

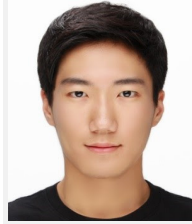
본 논문에서는 시간축에서 연속적인 임무 할당 검증 환경을 개발하였고 이 환경 하에서 사용할 수 있는 그리디 알고리즘을 제안하였다. 연속적인 환경이기에 언제나 임무를 중단하고 재개할 수 있는 만큼 계산과정이 복잡해져 임무와 로봇 수가 증가함에 따라 그리디 알고리즘의 한계는 명확해진다. 따라서 본 환경에 대한 심층 신경망 기반 학습 알고리즘의 필요성에 대해 생각해볼 수 있다. 다양한 수와 임무 수에 대해 대응할 수 있도록 입력값의 수가 자유로운 LSTM 및 트랜스포머 기반 심층 신경망 구조를 시도해볼 수 있다.

References

- [1] K. V. R. Devi, S. B. S. S. Lakhanpal, R. Kalra, V. A. Sethi, and S. K. Thajil, "A review: Swarm robotics: Cooperative control in multi-agent systems," *E3S Web of Conferences*, 2024, DOI: 10.1051/e3sconf/202450503013.
- [2] S. Han, Q. Zhang, Y. Yao, W. Jin, and Z. Xu, "LLM multi-agent systems: Challenges and open problems," *arXiv:2402.03578*, 2024, DOI: 10.48550/arXiv.2402.03578.
- [3] Q. Peng, H. Wu, and R. Xue, "Review of dynamic task allocation methods for UAV swarms oriented to ground targets," *Complex System Modeling and Simulation*, vol. 1, no. 3, pp. 163-175, Sept., 2021, DOI: 10.23919/CSMS.2021.0022.
- [4] J. Wu, J. Zhang, Y. Sun, X. Li, L. Gao, and G. Han, "Multi-UAV collaborative dynamic task allocation method based on ISOM and attention mechanism," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 5, pp. 6225-6235, May, 2024, DOI: 10.1109/TVT.2023.3341878.
- [5] M. Afrin, J. Jin, A. Rahman, A. Rahman, J. Wan, and E. Hossain, "Resource allocation and service provisioning in multi-agent cloud robotics: A comprehensive survey," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 842-870, 2021, DOI: 10.1109/COMST.2021.3061435.
- [6] Y. Jing, B. Liang, S. Li, F. Liu, W. Zhao, and P. Liu, "A multi-agent learning framework for mixed-integer linear programming," *INFOR: Information Systems and Operational Research*, vol. 62, no. 4, pp. 588-598, 2024, DOI: 10.1080/03155986.2024.2376446.
- [7] J. Cui, Y. Liu, and A. Nallanathan, "Multi-agent reinforcement learning-based resource allocation for UAV networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 2, pp. 729-743, 2019, DOI: 10.1109/TWC.2019.2935201.
- [8] C. Liu and A. Kroll, "A centralized multi-robot task allocation for industrial plant inspection by using A* and genetic algorithms," *Artificial Intelligence and Soft Computing*, pp. 466-474,

2012, DOI: 10.1007/978-3-642-29350-4_56.

- [9] M. Braquet and E. Bakolas, "Greedy decentralized auction-based task allocation for multi-agent systems," *arXiv:2107.00144*, 2021, DOI: 10.48550/arXiv.2107.00144.
- [10] K. Zhang, E. G. Collins, Jr., and D. Shi, "Centralized and distributed task allocation in multi-robot teams via a stochastic clustering auction," *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, vol. 7, no. 2, pp. 1-22, Jul., 2012, DOI: 10.1145/2240166.2240171.
- [11] Y. Chen, X. Mao, F. Hou, Q. Wang, and S. Yang, "Combining re-allocating and re-scheduling for dynamic multi-robot task allocation," *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Budapest, pp. 000395-000400, 2016, DOI: 10.1109/SMC.2016.7844273.
- [12] W. Dai, H. Lu, J. Xiao, Z. Zeng, and Z. Zheng, "Multi-robot dynamic task allocation for exploration and destruction," *Journal of Intelligent & Robotic Systems*, vol. 98, pp. 455-479, Oct., 2019, DOI: 10.1007/s10846-019-01081-3.



이재호

2021 한국과학기술원 항공우주공학과(학사)
2023 한국과학기술원 항공우주공학과(석사)
2023~현재 한국과학기술원 항공우주공학과(박사)

관심분야: 다중로봇 시스템, 휴리스틱 서치



홍민지

2022 한국항공대학교 항공우주 및 기계공학부(학사)
2024 한국과학기술원 항공우주공학과(석사)
2024~현재 한국과학기술원 항공우주공학과(박사)

관심분야: 다중로봇 시스템, 경로 계획, 임무 할당



김희연

2023 한국과학기술원 기계공학과(학사)
2025 한국과학기술원 김재철AI대학원(석사)
2025~현재 한국과학기술원 항공우주공학과(박사)

관심분야: 다중로봇 시스템, 강화학습



김성민

2024 한국항공대학교 항공우주 및 기계공학부(학사)
2024~현재 한국과학기술원 항공우주공학과(석사)

관심분야: 다중 로봇 시스템, 발사체 궤적 예측, 강화학습



최한림

2000 한국과학기술원 항공우주공학과(학사)
2002 한국과학기술원 항공우주공학과(석사)
2009 메사추세츠 공과대학교 항공우주공학(공학박사)
2008~현재 한국과학기술원 교수

관심분야: 다중로봇 시스템, 지능형 항공우주 시스템, 불확실성 정량화 및 학습