

논문 2022-17-15

# 충돌 회피가 보장된 분산화된 군집 UGV의 모델 예측 포메이션 제어

## (Distributed Model Predictive Formation Control of UGV Swarm Guaranteeing Collision Avoidance)

박성창, 이승목\*

(Seong-Chang Park, Seung-Mok Lee)

Abstract : This paper proposes a distributed model predictive formation control algorithm for a group of unmanned ground vehicles (UGVs) with guaranteeing collision avoidance between UGVs. Generally, the model predictive control based formation control has a disadvantage in that it takes a long time to compute control inputs when considering collision avoidance between UGVs. In this paper, in order to overcome this problem, the formation control algorithm is implemented in a distributed manner so that it could be individually controlled. Also, a collision-avoidance method considering real-time is proposed. The proposed formation control algorithm is implemented based on robot operating system (ROS), open source-based middleware. Through the various simulation tests, it is confirmed that the formation control of five UGVs is successfully performed while avoiding collisions between UGVs.

Keywords : Formation Control, Model Predictive Control, UGV (Unmanned Ground Vehicle), ROS (Robot Operating System).

### 1. 서론

넓은 공간에 대한 탐색 및 감시와 같은 임무를 수행함에 있어서 단일 Unmanned Ground Vehicle (UGV) 시스템 보다 다수의 UGV가 군집을 이루어 분산적으로 임무를 수행하는 것이 훨씬 효율적이다. 다수 UGV가 공간 탐색 및 감시 임무를 수행하기 위해서는 포메이션 제어 기술이 필수적으로 요구된다. 포메이션 제어 기술은 각 UGV가 자신의 위치와 주변 UGV들의 위치를 파악하여 서로 상대적인 위치를 일정하게 유지하면서 이동하는 기술이다.

군집 모바일 로봇 포메이션 제어에 대한 연구는 다양한 응용분야에서 활발히 연구되어 오고 있다. 미국 MIT CSAIL 연구소 [1]은 매니플레이터를 장착한 전방향 모바일 로봇을 이용하여 물체 운송을 위한 포메이션 제어 기법과 전역 경로계획, 지역 경로계획 등의 기반 기술을 통합하여 동적인 환경에서 운용 가능한 군집 운송 로봇 시스템을 개발하였다. 유럽 Univ. of Zaragoza [2]에서는 타겟의 추적 및 포위를 위한 적응형 포메이션 제어 알고리즘을 개발하였다. 타겟을 효과적으로 포위하기 위한 포메이션을 상황에 따라 변환이 가능하도록 제어한다.

최근에는 군집 UGV 포메이션 제어를 모델 예측 제어 (Model Predictive Control) 기법을 기반으로 해결하려는 시도가 많이 이루어지고 있다. 모델 예측 제어를 포메이션 제어 문제에 적용하게 되면 UGV 간의 충돌 회피 조건을 구속 조건으로 설정하여 충돌 회피 문제를 해결 할 수 있다는 장점이 있다 [3]. 하지만 구속조건을 만족하는 해를 찾지 못하는 경우가 발생 가능하고 실시간성을 보장할 수 없다는 문제점이 있다.

이러한 문제를 해결하기 위해 참고논문 [4]에서는 충돌 회피 구속 조건을 만족하는 준최적해를 빠르게 찾는 Receding Horizon Particle Swarm Optimization (RHPSO) 기반의 모델 예측 포메이션 제어 기법을 개발하였다. RHPSO 기법은 메타휴리스틱 최적화 기법 중 하나로 다수의 후보 해 중에서 충돌 회피 구속 조건을 만족하는 후보 해만을 선택하여 그 중에서 최소의 최적해를 찾는 기법이다. 다만 드물게 모든 후보 해가 구속조건을 만족하지 않는 경우가 발생할 수 있어 충돌 회피를 완전히 보장할 수 없다는 한계가 있다. 또한 UGV 간 통신 네트워크 토폴로지 구조에 따라 하위 레벨에 위치한 로봇은 상위 레벨에 위치한 모든 UGV들과의 충돌 회피 여부를 검사해야하기 때문에 하위 레벨에 위치한 UGV일수록 계산 시간이 증가한다는 문제점이 있다.

참고문헌 [5]에서는 RHPSO 기법에 대하여 최적화 과정에서 사용되는 후보 해의 수, 반복 연산 회수에 따른 성능을 분

\*Corresponding Author (seungmok@sch.ac.kr)

Received: Sep. 2, 2021, Revised: Oct. 15, 2021, Accepted: Nov. 17, 2021.

S.C. Park: Soonchunhyang University (M.S. Student)

S.M. Lee: Soonchunhyang University (Prof.)

\* 본 연구는 방위사업청과 국방과학연구소가 지원하는 군집형 무인 CPS 특화연구실 사업의 일환으로 수행되었습니다 (UD190029ED).

석하였으며, 충분히 적은 연산량으로 허용가능한 제어 오차를 보여줌으로써 실시간 제어에 적용 가능성을 확인하였다.

참고문헌 [6]에서는 두 대의 드론을 이용한 편대 비행 제어 시스템을 제안하였으며, 지상 제어 시스템과의 실시간 통신을 통해 임무 전달 및 제어하는 방법을 채택하였다. 참고문헌 [7]에서는 RTK-GPS 기반의 정밀 위치인식을 바탕으로 실외 환경에서의 드론 군집 비행 시스템을 개발하였다. 두 방법 모두 지상 서버를 통해 제어 명령을 전달하는 중앙 집중형 제어 방식을 채택하고 있다.

UGV 포메이션 제어 알고리즘이 실제 필드에 적용되기 위해서는 중앙 서버에서 제어 명령을 계산하여 전송하는 중앙 집중형 방식이 아닌 각각의 UGV에서 개별적으로 제어 명령을 생성하여 움직이는 분산형 방식으로 구현되어야 한다. 중앙 집중형 방식의 경우 군집 UGV가 서버로부터 멀리 이동하거나 서버와의 통신을 담당하는 UGV가 고장 나는 경우 군집 UGV는 더 이상 임무를 수행하기 어려워지기 때문이다. 이러한 문제를 해결하기 위해서는 본 논문에서는 각 UGV가 주변 UGV들의 움직임을 고려하여 스스로 판단하여 움직일 수 있도록 포메이션 제어 알고리즘을 분산화하여 구현하였다.

본 논문에서는 군집 UGV 시스템의 분산화된 포메이션 제어 알고리즘을 오픈 소스 기반의 미들웨어인 Robot Operating System (ROS)을 이용하여 구현하였다. 물리엔진 기반 3D 로봇 시뮬레이터인 Gazebo와 연동하여 실제 환경과 매우 유사한 시뮬레이터를 구축하였다. 다양한 포메이션 제어 시나리오를 구성하여 본 연구를 통해 개발된 분산화된 편대 제어 알고리즘의 실시간성을 테스트하였다. 기존 연구 대비 본 논문을 통해 제안된 모델 예측 포메이션 제어 기법은 다음과 같은 차별성을 가진다. 첫째, 포메이션 전환 시 UGV 간의 충돌이 발생하지 않음을 보장한다. 둘째, 계산 시간이 작으면서 항상 일정하게 유지하여 제어 알고리즘의 실시간성을 보장한다. 셋째, 다수의 UGV에 포메이션 제어 알고리즘을 각각의 UGV 내부에서 개별적으로 제어 알고리즘이 동작하도록 완전히 분산화하여 구현하였다. 본 연구에서는 5대의 저전력의 소형 임베디드 컴퓨팅 보드에 제안된 포메이션 제어 알고리즘을 분산화하여 구현하고, UGV 간의 충돌 회피와 실시간성을 검증하였다.

본 논문의 구성은 다음과 같다. II장에서는 충돌 회피가 고려된 UGV 포메이션을 최적화 문제로 정의하고, 모델 예측 제어 기반의 포메이션 최적 제어 기법에 대하여 소개한다. III장에서는 UGV 간의 충돌 회피에 대한 구속 조건을 항상 만족시키는 해를 찾는 기법을 제안하고, 이를 ROS 기반의 분산화된 포메이션 제어 기법의 구현에 대하여 다룬다. IV장에서는 UGV 간의 충돌이 발생할 수 있는 다양한 시나리오에 대한 실시간 시뮬레이션을 통해 제안된 기법의 성능을 평가한다. 그리고 V장에서는 결론을 통해 제안된 기법의 효과를 요약한다.

## II. 포메이션 문제 정의 및 한계점

### 1. 군집 UGV 포메이션 제어 문제 정의

모델 예측 제어 기반의 군집 UGV 포메이션 제어는 다음과 같이 최적화 문제로 정의될 수 있다. 포메이션 형성 및 유지력을 위한 UGV  $i$ 에 해당하는 포메이션 제어 오차를  $e_i(t)$ , 제어 입력을  $u_i(t)$ 라고 할 때, 시간  $t$ 에서의 UGV 간 충돌 회피를 고려한 포메이션 제어 입력  $u_i(t)$ 는 다음의 최적화 문제를 통해 계산된다.

$$\min_{u_i} J_i(e_i(t), u_i(t)), \quad (1)$$

$$d_{ij}(\tau) \geq d_{safe}, \forall j \in \eta_i. \quad (2)$$

식 (2)는 UGV 간 충돌 회피를 위한 구속 조건으로  $d_{ij}$ 는 UGV  $i$ 와 UGV  $j$  간의 거리를 나타내며  $d_{safe}$ 는 UGV 간 유지해야 하는 최소한의 거리,  $\eta_i$ 는 UGV  $i$ 와 인접한 모든 UGV들의 인덱스 집합을 의미한다. 비용 함수  $J_i$ 는 예측 구간  $T$ 에 대하여 다음과 같이 정의된다.

$$J_i(e_i(t), u_i(t)) = \int_t^{t+T} (e_i^T(\tau)Qe_i(\tau) + u_i^T(\tau)Ru_i(\tau))d\tau. \quad (3)$$

포메이션 제어 오차  $e_i(t)$ 는 주변 UGV들과의 상대 위치와 포메이션 명령 패턴에 의해 결정된다 [4]. 행렬  $Q$ 와  $R$ 은 각 상태 변수의 가중치를 나타내는 양정치 행렬이다.

군집 UGV가 초기 임의의 위치에서 명령 포메이션을 형성하거나 다른 포메이션으로 전환 시 UGV들은 움직이는 경로가 서로 겹치면서 개체 간 충돌의 위험성이 커지게 된다. UGV들 간의 충돌을 회피하기 위해서는 개체 간 거리가 항상 일정 거리 이상으로 유지되어야 한다. 이러한 조건은 구속 조건 (2)로 표현될 수 있으며, UGV들 간의 충돌을 회피하면서 포메이션이 명령에 따라 제어되기 위해서는 구속조건 (2)을 만족시키면서 식 (3)의 비용 함수  $J_i$ 를 최소화하는 제어 입력  $u_i(t)$ 를 찾아야 한다.

### 2. 기존 RHPSO 기법의 한계점

식 (1)-(3)에서 정의된 최적화 문제를 효과적으로 해결하기 위해 RHPSO 기법 [4]이 제안되었다. 이동 구간 입자 군집 최적화 알고리즘은 다수의 후보 해를 생성하여 메타 휴리스틱 기반의 반복적인 연산을 통해 준최적해를 빠르게 찾는다. 각 후보 해를 나타내는 입자는 시간  $t_k$ 에서 계산된 예측 구간  $T$  동안의 제어 입력  $u_i(t_k) = [v_i(t_k), \omega_i(t_k)]^T$ 의 시계열을 나타내며 상세한 표현형은 다음과 같다.

$$[v_i(t_{k+1}; t_k), v_i(t_{k+2}; t_k), \dots, v_i(t_{k+N}; t_k), \omega_i(t_{k+1}; t_k), \omega_i(t_{k+2}; t_k), \dots, \omega_i(t_{k+N}; t_k)] \quad (4)$$

충돌 회피를 위해 다수의 후보 해 중에서 구속조건 (2)을 만족시키는 해 중 비용함수 값이 가장 최소인 해를 UGV의 제어 입력으로 사용한다. 이러한 방법을 통해 구속조건 (2)를 만족시키는 준최적해를 쉽게 찾을 수 있다는 장점이 있다.

하지만 위 방법은 후보 해 중에 구속조건 (2)을 만족시키는 해가 하나도 존재하지 않는 경우가 드물게 발생한다는 문제점이 있다. 이러한 경우 생성된 제어 입력이 실제 UGV에 적용될 경우 충돌이 발생할 가능성이 있어 RHPSO 방법은 충돌 회피를 완전히 보장할 수 없다는 한계가 있다.

### III. 제안된 군집 UGV 포메이션 제어 기법

#### 1. UGV 간 충돌 회피를 고려한 최적화 기법

본 장에서는 기존 RHPSO 방법을 기반으로 군집 UGV 포메이션 제어 시 개체 간 충돌이 감지되었을 때 UGV 간 통신 네트워크 토폴로지에 관계없이 계산 시간이 일정하면서 충돌 회피를 보장하는 방법을 제시한다. UGV 간 충돌 감지는 예측 구간  $t_{k+1}$  에서  $t_{k+N}$  까지 timestep 별로 UGV 간 거리를 계산하여 안전거리 보다 작으면 충돌이 감지된 것으로 간주한다. 그림 1 은 두 대의 UGV 간 충돌이 감지된 경우에 대한 예시를 보여준다. 그림에서와 같이 UGV 1 과 UGV 2는  $t_{k+c}$  에서 UGV 간 거리가 안전거리보다 가까워져 충돌이 감지되었다고 가정하자. 이 경우는 입자 군집 최적화 과정을 통해 충돌 회피 조건을 만족하는 해를 찾지 못한 경우이다. 이러한 해를 실제 UGV에 적용할 경우 UGV 간 충돌이 발생할 확률이 커진다.

이러한 문제를 해결하기 위해 기존 RHPSO 방법을 통해 최종적으로 찾은 해가 충돌 회피 조건을 만족하지 않는 경우 UGV가 감속을 하도록 하여 강제적으로 수정하여 충돌을 피하도록 한다. 예측 구간  $[t_{k+1}, t_{k+2}, \dots, t_{k+N}]$  내에서 충돌이 감지된 timestep을  $t_{k+c}$  라 할 때, UGV 감속을 위한 선속도  $\hat{v}_i$  값은 식 (5)를 따른다.

$$\hat{v}_i(t_{k+j}; t_k) = \left(\frac{c}{N}\right)v_i(t_{k+j}; t_k), \forall j \in \{1, \dots, N\} \quad (5)$$

제안된 충돌 방법은 충돌이 예측되는 두 UGV 사이에 사전에 정의된 UGV 간 계층적 구조의 통신 네트워크 구조에 따라 하위 레벨에 있는 UGV가 선속도 명령 값을 수정하여 충돌을 회피하는 방법이다. 예측구간 내에 충돌이 감지되면 감지된 timestep을 기준으로 UGV의 선속도를  $c/N$  의 비율로 감소시킨다. 일반적인 경우 UGV들이 감속하면서 움직이는 과정에서 충돌 지점까지 접근하기 전에 충돌이 해소되며 다시 원래의 속도를 회복하여 움직이게 된다. 충돌이 끝까지 해소되지 않고 충돌 지점까지 접근하게 되는 극단적인 경우, 그림 2와 같이 계산된 경로 위의 충돌이 예측되는 지점까지

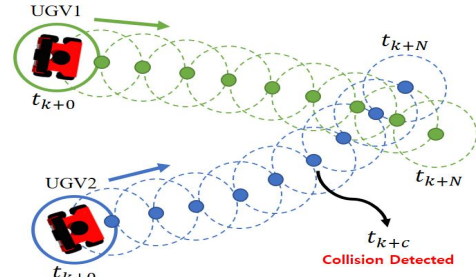


그림 1. 예측 구간 내에 UGV 간 충돌이 감지된 경우에 대한 예시  
Fig. 1. A case illustration in which a collision between UGVs is detected within the prediction horizon

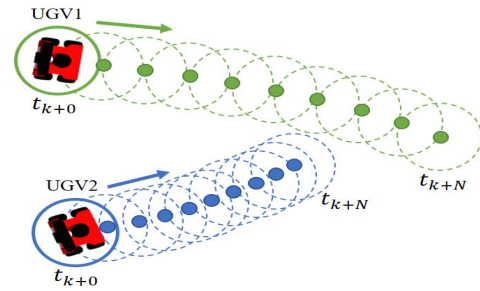


그림 2. 제안된 방법에 의해 UGV 간 충돌이 회피된 경우에 대한 예시

Fig. 2. A case illustration in which a collision between UGVs is avoided by the proposed method

접근하지 않고 그 전에 감속을 통해 충돌 지점에서 정지하게 되어 충돌을 회피할 수 있다. 따라서 식 (5)에 의한 감속은 충돌이 감지된 지점까지 UGV가 접근하지 못하도록 하여 UGV 간 충돌 회피를 보장할 수 있다.

#### 2. ROS 기반 분산화된 포메이션 제어 구현

본 논문에서는 군집 UGV 포메이션 제어 알고리즘을 분산화하여 각 UGV에서 개별적으로 제어 알고리즘이 구동될 수 있도록 구현하는데 초점을 맞추고 있다. 제안된 알고리즘의 구현 및 실험을 위해 오픈 소스 기반의 미들웨어인 ROS를 이용하여 구현하였으며, 포메이션 제어 알고리즘의 분산화를 위해 각 UGV의 포메이션 제어 알고리즘을 별도의 ROS 노드로 구현하였다. 이때 사용되는 UGV 통신 네트워크 토폴로지는 그림 3과 같이 군집 포메이션 제어에서 일반적으로 많이 사용되는 계층적 구조를 갖는 토폴로지를 사용하였다. 즉, 서버에서 군집 UGV가 이동해야 하는 기준 경로와 포메이션 패턴 명령을 생성하여 선도 UGV (UGV 1)에 게 전달한다. 선도 UGV는 전달받은 정보를 바탕으로 자신의 이동 경로를 최적화한다. 그리고 자신의 이동 경로 정보를 네트워크 토폴로지의 하위 레벨에 위치한 UGV들에게 전송하여 이를 바탕으로 추종 UGV들이 자신의 이동 경로를 최적화 할 수 있도록 한다.

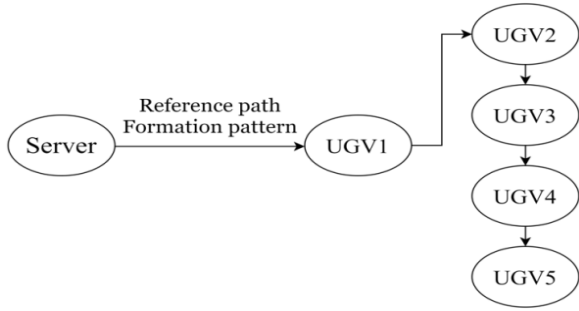


그림 3. ROS 노드의 계층적 통신 구조

Fig. 3. Hierarchical communication structure of ROS nodes

기존 RHPSO [4] 기법을 그림 3과 같은 계층적 통신 네트워크 토폴로지 상에서 구현할 경우 각 UGV의 계층적 위치에 따라 계산 시간이 달라질 수 있다는 문제점이 있다. 기존 RHPSO 기법은 충돌 회피 구속조건을 만족시키기 위해 모든 후보 해에 대하여 충돌 회피 여부를 검사해야 하는데, 이 경우 네트워크 토폴로지상에서 하위 레벨에 위치해 있는 UGV는 자신의 상위 레벨에 위치한 모든 UGV에 대하여 충돌 회피 여부를 검사해야 한다. 따라서 하위 레벨에 위치한 UGV 일수록 충돌 회피를 위한 계산 시간이 길어진다는 문제점이 있다.

하지만 본 논문에서 제안된 알고리즘의 경우 모든 후보 해에 대하여 충돌 회피 여부를 검사하지 않고 최적화 알고리즘을 통해 최종적으로 계산된 최적해에 대해서만 검사하여 충돌이 감지된 경우 식 (5)를 통해 속도를 수정한다. 따라서 계산 시간을 크게 줄이면서 효과적인 충돌 회피가 가능하다는 장점이 있다.

#### IV. 실험 결과

##### 1. 실험 환경

분산화된 포메이션 제어 알고리즘의 성능 시험을 위해 총 5대의 UGV를 가정하여 실시간 시뮬레이션이 가능하도록 실험 환경을 구축하였다. 그림 4와 같이 물리 엔진 기반 3D 시뮬레이터 Gazebo 구동을 위한 서버 PC, 각 UGV의 분산 포메이션 제어 알고리즘 구현을 위한 5대의 임베디드 보드, UGV 간 통신을 위한 무선 라우터를 사용하였다. UGV 간 통신은 무선 라우터를 통해 그림 3 과 같은 계층적인 통신 네트워크 구조 상에서 TCP/IP 통신으로 정보를 주고받는다. 임베디드 보드는 NVIDIA 사의 Nano Jetson [8]을 사용하였다. 각 UGV들은 무선 TCP/IP 통신을 통해 정보를 주고받으면서 5 Hz의 속도로 최적화 과정을 수행하고 각 UGV의 제어 입력을 분산적으로 계산한다. 각 임베디드 보드에서 계산된 UGV의 속도 제어를 위한 명령은 서버 PC로 보내져서 Gazebo의 3D 가상 UGV를 구동하여 각 UGV의 움직임을 실시간으로 확인할 수 있다.

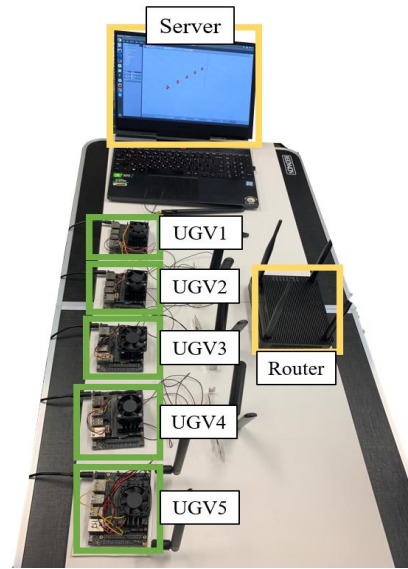


그림 4. 실시간 포메이션 제어 실험을 위해 5대의 분산화된 임베디드 컴퓨팅 보드, 서버 PC, 무선 TCP/IP 통신으로 구성된 실험환경 구축

Fig. 4. Experimental setup for real-time formation control test consisting of five distributed embedded computing boards, a server PC, and wireless TCP/IP communication

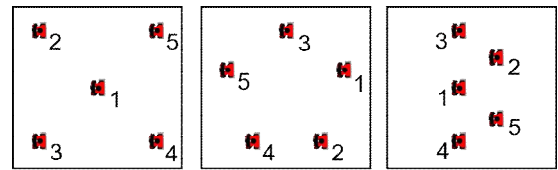


그림 5. 명령 포메이션 패턴 (왼쪽에서부터 F1, F2, F3)

Fig. 5. Command formation pattern (F1, F2, F3 from left side)

제안된 포메이션 제어 알고리즘의 UGV 간 충돌 회피 성능을 시험하기 위해 그림 5와 같이 UGV 5대에 대한 포메이션 패턴을 사전에 설정해두고 시간에 따라 포메이션을 순차적으로 변경하도록 명령을 주었다. 포메이션 변경 시 UGV들은 서로의 위치를 변경하는 과정에서 충돌이 일어날 가능성이 높아지며, 이러한 실험을 통해 충돌 발생 여부를 확인할 수 있다.

본 연구에서는 포메이션 변경에 대한 3개의 Test 시나리오를 설정하였다. Test 1은 F1 → F2 변경, Test 2는 F2 → F3 변경, Test 3은 F3 → F1 변경이 수행되는 3개의 시나리오를 설정하였으며, 제시된 포메이션 변경 수행을 위해서는 UGV들의 충돌 가능성이 높기 때문에 충돌 회피가 반드시 이루어져야 하는 시나리오이다.

본 논문에서 제안된 실험 환경은 각 임베디드 보드에 UGV 포메이션 제어 알고리즘을 분산적으로 구현하여 실시간 시뮬레이션을 수행하기 때문에 실제 실험과 매우 유사한 환경이라고 볼 수 있다.

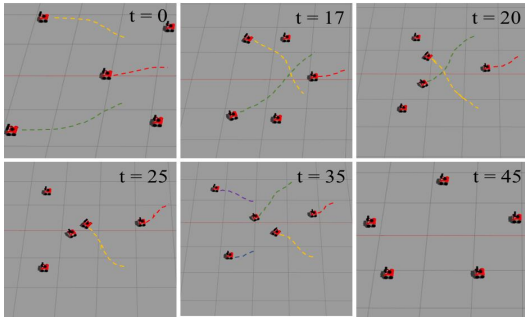


그림 6. Test 1 (F1→F2)의 Gazebo 시뮬레이터 스냅샷  
Fig. 6. Gazebo simulator snapshots of Test 1 (F1→F2)

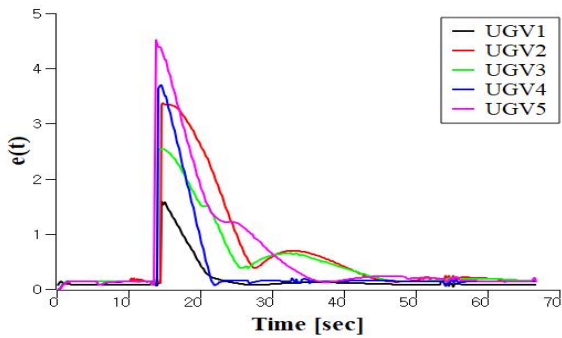


그림 7. Test 1 (F1→F2)에 대한 각 UGV들의 포메이션 에러  
Fig. 7. Formation errors of each UGV for Test 1 (F1→F2)

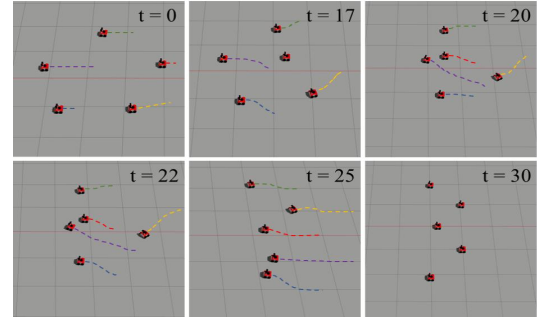


그림 8. Test 2 (F2→F3)의 Gazebo 시뮬레이터 스냅샷  
Fig. 8. Gazebo simulator snapshots of Test 2 (F2→F3)

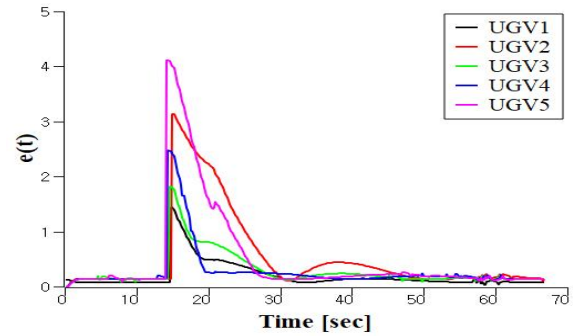


그림 9. Test 2 (F2→F3)에 대한 각 UGV들의 포메이션 에러  
Fig. 9. Formation errors of each UGV for Test 2 (F2→F3)

2. 실험 결과 및 분석

그림 6은 Test 1에 대한 Gazebo 시뮬레이터 실시간 스냅샷을 보여주고 있다. 그림 6에서 볼 수 있듯이 사전에 설정된 포메이션 변경 시나리오에 따라 포메이션 패턴 F1에서 F2로 성공적으로 변경이 되는 것을 확인할 수 있다.

그림 7은 Test 1에 대한 각 UGV들의 시간에 따른 포메이션 에러를 보여주고 있다. UGV가 움직이기 시작한 후 15초 후에 포메이션 패턴 명령이 내려졌기 때문에 포메이션 에러가 증가한 것을 확인할 수 있으며, 시간이 지남에 따라 포메이션 에러가 0에 가깝게 수렴하여 포메이션 패턴 F2로 성공적으로 변경이 되었음을 확인할 수 있다. 다만, 본 실험에서는 무선 통신 지연, 로봇의 구동 오차 등이 모두 그대로 구현되어 실제 실험 환경과 매우 유사한 환경에서 실험이 진행되었기 때문에 에러가 0으로 완전히 수렴하지 않은 것으로 판단된다.

같은 방법으로 Test 2와 Test 3에 대한 실험을 진행하였으며, 그 결과는 그림 8-11에서 확인할 수 있다. Test 2와 Test 3의 경우에도 주어진 시나리오에 따라 포메이션 변경이 성공적으로 이루어졌음을 확인할 수 있다.

그림 12-14는 각 Test에 대하여 UGV들의 이동 속도를 보여준다. 포메이션의 이동 속도인 0.3 m/s의 속도를 유지하다가 포메이션 패턴 변경 명령이 내려진 15초 이후 일부 로

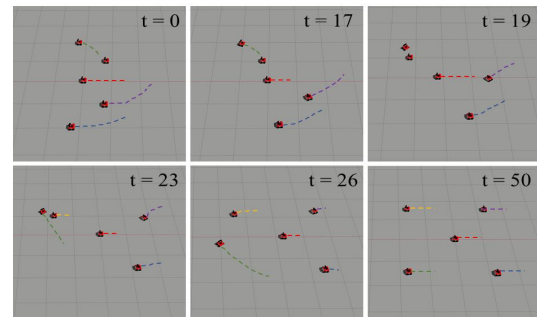


그림 10. Test 3 (F3→F1)의 Gazebo 시뮬레이터 스냅샷  
Fig. 10. Gazebo simulator snapshots of Test 3 (F3→F1)

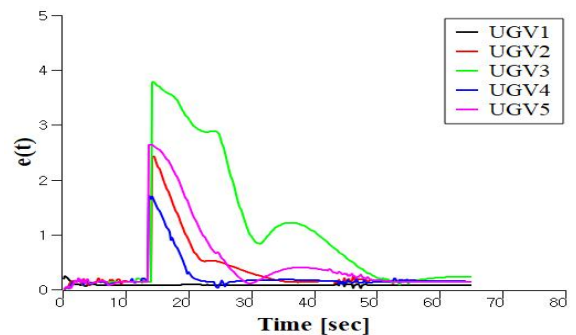


그림 11. Test 3 (F3→F1)에 대한 각 UGV들의 포메이션 에러  
Fig. 11. Formation errors of each UGV for Test 3 (F3→F1)

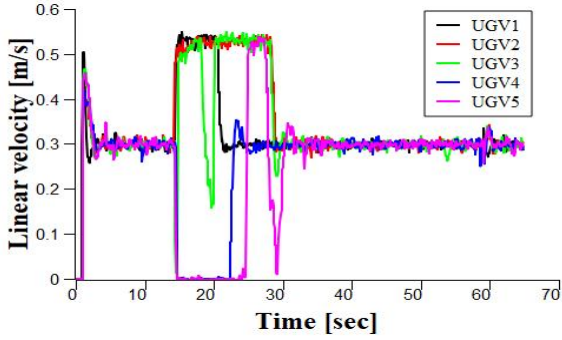


그림 12. Test 1 (F1→F2)에 대한 각 UGV들의 이동 속도  
Fig. 12. Linear velocity of each UGV for Test 1 (F1→F2)

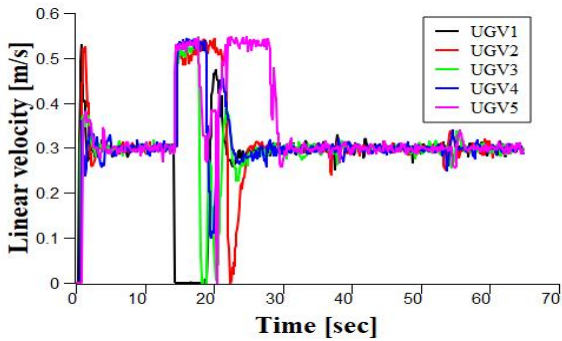


그림 13. Test 2 (F2→F3)에 대한 각 UGV들의 이동 속도  
Fig. 13. Linear velocity of each UGV for Test 2 (F2→F3)

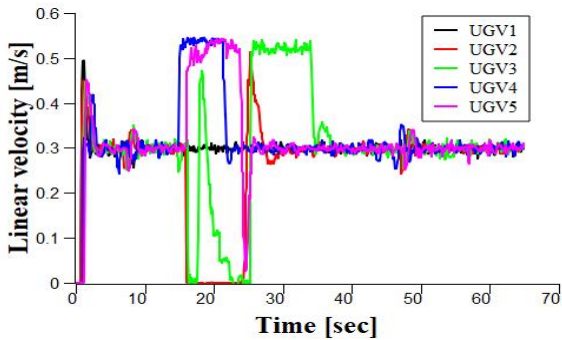


그림 14. Test 3 (F3→F1)에 대한 각 UGV들의 이동 속도  
Fig. 14. Linear velocity of each UGV for Test 3 (F3→F1)

봇들이 충돌을 회피하기 위한 이동 속도의 감속이 이루어지는 것을 확인할 수 있다.

제안된 포메이션 제어 알고리즘의 UGV 간 충돌 회피 성능을 확인하기 위해 앞에서 제안된 Test 시나리오에 대하여 기존 RHPSO 방법과 본 논문에서 제안된 방법의 충돌 여부를 시험하였다. RHPSO 기법은 확률을 기반으로 연산 때문에 실행할 때 마다 결과 값이 달라진다는 특성을 갖고 있다. 따라서 각 Test 시나리오에 대하여 30번 반복하여 충돌 여부를 확인하였다.

표 1. 각 Test 시나리오별 30번 반복 실험 중 UGV 간 충돌이 발생한 횟수 (충돌 회피 성공 비율)

Table 1. Number of collisions between UGVs during 30 runs for each test scenario (Collision avoidance success rate)

	Conventional	Proposed
Test 1	8 (73%)	0 (100%)
Test 2	3 (90%)	0 (100%)
Test 3	24 (20%)	0 (100%)

표 2. 기존 방법과 제안된 방법의 계산 시간

Table 2. Computation time between the conventional and proposed methods

	Conventional	Proposed
UGV1	0.0914	0.0672
UGV2	0.1243	0.0655
UGV3	0.1749	0.0666
UGV4	0.2335	0.0675
UGV5	0.2877	0.0671

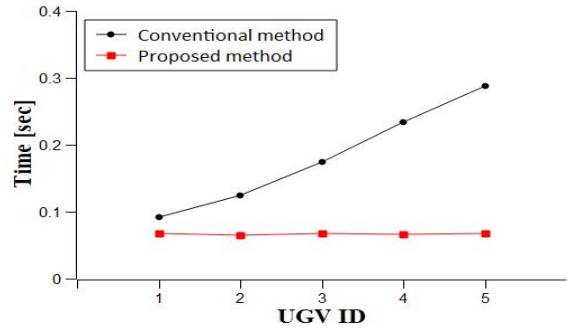


그림 15. 기존 방법과 제안된 방법의 계산 시간 비교  
Fig. 15. Comparison of computation time between the conventional and proposed methods

각 Test 에 대한 충돌 횟수는 표 1에서 확인할 수 있다. 기존 알고리즘은 충돌 회피 조건을 만족하는 해를 찾지 못하는 경우가 발생할 가능성이 있기 때문에 제시된 Test 시나리오와 같이 UGV 간 충돌 가능성이 매우 높은 조건에서는 실제 충돌이 발생할 수 있다. 하지만 제안된 알고리즘의 경우 30번 반복 실험을 하는 동안 모든 실험에 대하여 충돌이 발생하지 않음을 확인할 수 있다. 따라서 본 논문에서 제안한 알고리즘은 분산화된 환경에서 UGV들 간 충돌 회피를 완전히 보장할 수 있다는 것을 실험적으로 확인하였다.

마지막으로 제안된 알고리즘의 실시간성을 확인하기 위해 본 논문에서 사용된 NVIDIA Nano Jetson 보드에서의 연산 시간을 측정하였으며, 그 결과는 표 2와 그림 15와 같다. 본 실험에서는 그림 3의 계층적 구조를 갖는 통신 네트

워크 구조를 갖기 때문에 기존 알고리즘의 경우 하위 레벨에 위치한 UGV 일수록 계산시간이 점점 증가한다는 것을 확인할 수 있다. 반면 제안된 방법의 경우 통신 네트워크 토폴로지 구조와 관계없이 연산 시간이 일정하게 유지되는 것을 확인할 수 있다.

**V. 결론**

본 논문에서는 군집 UGV의 포메이션 제어를 위해 ROS 환경에서 소프트웨어 및 하드웨어를 분산화시켜 포메이션 제어 알고리즘을 구현하고, 포메이션 변경 시 UGV 간 충돌 회피가 보장되는 방법을 제안하였다. 임베디드 보드와 물리 엔진 기반의 Gazebo 시뮬레이터를 활용하여 실제 환경과 매우 유사한 실시간 시뮬레이터를 구축하였다. 임베디드 보드 기반 실시간 시뮬레이션을 통해 사전에 설정된 시나리오에 따라 5대의 UGV들의 포메이션 제어가 성공적으로 이루어짐을 확인하였다. 또한 명령에 따라 포메이션 변경을 수행할 때 UGV 간 충돌 회피가 보장됨을 확인하였으며, UGV 간 통신 네트워크 토폴로지에 관계없이 계산 시간이 일정하다는 것을 확인하였다.

향후 연구에서는 실제 5대의 UGV 하드웨어에 본 논문에서 제안한 분산화된 포메이션 제어 알고리즘을 적용하여 필드 테스트를 수행할 계획이며, 실제 적용 시 발생할 수 있는 노이즈에 대한 강건성을 확인할 예정이다.

**References**

[1] J. Alonso-Mora, S. Baker, D. Rus, "Multi-robot Formation Control and Object Transport in Dynamic Environments Via Constrained Optimization," *International Journal of Robotics Research*, Vol. 36, No. 9, pp. 1000-1021, 2017.

[2] G. López-Nicolás, M. Aranda, Y. Mezouar, "Adaptive Multirobot Formation Planning to Enclose and Track a Target with Motion and Visibility Constraints," *IEEE Transactions on Robotics*, Vol 36, No. 1, pp. 142-156, 2020.

[3] A. S. Lafmejani, S. Berman, "Nonlinear MPC for Collision-free and Deadlock-free Navigation of Multiple Nonholonomic Mobile Robots," *Robotics and Autonomous Systems*, Vol. 141, 103774, 2021.

[4] S. M. Lee, H. Myung, "Receding Horizon Particle Swarm Optimisation-based Formation Control with Collision Avoidance for Nonholonomic Mobile Robots," *IET Control Theory & Applications*, Vol. 9, No. 14, pp. 2075-2083, 2015.

[5] S. M. Lee, "Statistical Analysis of Receding Horizon Particle Swarm Optimization for Multi-robot Formation

Control," *Journal of the Korea Industrial Information Systems Research*, Vol. 24, No. 5, pp. 115-120, 2019 (in Korean).

[6] D. J. Kim, Y. S. Park, "An Implementation of Formation Flight Control System Using two Drones," *IEMEK J. Embed. Sys. Appl.*, Vol. 11, No. 6, pp. 343-351, 2016 (in Korean).

[7] S. Moon, Y. Choi, D. Kim, M. Seung, H. Gong, "Outdoor Swarm Flight System Based on RTK-GPS," *Korean Institute of Information Scientists and Engineers*, Vol. 43, No. 12, pp. 1315-1324, 2016 (in Korean).

[8] <https://developer.nvidia.com/embedded/jetson-nano-developer-kit/>

**Seong-Chang Park (박 성 창)**



2021 Dept. of Electronic Engineering from Keimyung University (B.S.)  
2021~Dept. of Future Convergence Technology from Soonchunhyang University (M.S.)

Field of Interests: Multi-agent system, formation control, simultaneous localization and mapping (SLAM)  
Email: scpark@sch.ac.kr

**Seung-Mok Lee (이 승 목)**



2006 Dept. of Physics from Chung-Ang University (B.S.)  
2008 Dept. of Satellite Systems and Applications Engineering from UST (M.S.)  
2014 Dept. of Civil and Environmental Engineering from KAIST (Robotics Program) (Ph.D.)

**Career:**

2015 Urban Robotics Laboratory from KAIST (Post-doctoral fellow)  
2017 Hyundai Motor Company (Senior Research Engineer)  
2021 Dept. of Automotive System Engineering from Keimyung University (Assistant Professor)  
2021~Dept. of Smart Automobile from Soonchunhyang University (Assistant Professor)  
Field of Interests: Autonomous vehicles, simultaneous localization and mapping (SLAM), soft computing, artificial intelligence (AI).  
Email: seungmok@sch.ac.kr