

다중 드론 기반 비행 애드혹 네트워크를 위한 개선된 네트워크 자가복구 기술

김동혁*, 백호기*, 임재성^o

Improved Network Self-Recovery Technique for Multi-Drone Based Flying Ad Hoc Networks

Donghyeok Kim*, Hoki Baek*, Jaesung Lim^o

요약

최근 드론과 같은 비행체를 활용한 Flying Ad Hoc Network (FANET) 기술이 주목을 받고 있다. 특히, 단일 드론이 아닌 다중 드론을 하나의 그룹으로 묶어 서비스를 지원하는 다중 드론 기반 FANET에 대한 연구가 활발히 진행되고 있다. 다중 드론 기반 FANET은 다중홉으로 연결되어 서비스 커버리지를 넓힐 수 있고, 특정 드론에 장애가 발생하더라도 다른 드론이 임무를 대체하여 신뢰성 있는 서비스를 지원할 수 있다. 이러한 다중 드론 기반 FANET 환경에서 가장 중요한 것은 모든 드론이 직접 혹은 다중홉을 통해 지상 제어 센터(GCS: Ground Controls System)에 연결되어야 한다는 점이다. 그러나 드론의 이동성 때문에 지상 제어 센터나 이웃 드론과의 연결이 끊어지면, 드론이 고립되고 네트워크가 단절되는 문제가 발생할 수 있다. 본 논문에서는 다중 드론 기반의 FANET 환경에서 드론이 스스로 링크 단절을 신속하게 인지하고 네트워크를 자가 복구할 수 있는 기술을 제안한다. 제안 기술은 지상 제어 센터로부터의 홉 수(hop count) 정보를 사용하여 드론 스스로 단절 여부를 파악하고, 단절 의심 단계를 거쳐 단절이 확정되면 제안하는 자가 복구 알고리즘을 통해 네트워크를 복구한다. 제안된 기법의 성능 평가를 위해 시뮬레이션을 하였으며, 결과 그래프를 통해 제안된 기법의 효과성을 입증하였다.

Key Words : Network disconnection, Multi-drone networks, Multi-Hop, UAV, Self-Recovery

ABSTRACT

Recently, flying ad hoc network (FANET) using drone-like vehicles has been attracting attention. In particular, research on multi-drone-based FANET that supports service by grouping multi-drones rather than single drone is actively performed. The multi-drones-based FANETs can broaden service coverage through multi-hop and, in the event of specific drone failure, other drones can replace the mission to support reliable services. In this multi-drone-based FANET, the most important thing is that all drones must be connected to ground controls system (GCS) directly or through multi-hop. However, if the connection is cut off due to the drones' mobility, the problems of isolating drones and network disconnection may arise. In this paper, we propose a technique to enable drone to recognize link disconnection quickly and to recover the network in the multi-drone based FANET environment. The proposed technique uses hop count information from GCS to identify whether the

* 본 연구는 2016년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. 2016R1A2A1A05005541).

• First Author : Ajou University, Department of Military Digital Convergence, gur5286@naver.com, 학생회원

o Corresponding Author : Department of Software, Ajou University, jaslim@ajou.ac.kr, 종신회원

* Department of Military Digital Convergence, Ajou University, neloyou@ajou.ac.kr, 정회원

논문번호 : 201903-020-B-RU, Received March 15, 2019; Revised March 22, 2019; Accepted March 24, 2019

drones are disconnected by themselves, and if the disconnection is confirmed through the disconnection step, the network is restored by the proposed self recovery algorithm. The performance is simulated and the effectiveness of the proposed method is verified through the results.

I. 서 론

드론은 초기 군사용으로 개발되다가 최근 상업용으로 개발되어 다양한 영역에서 확대되고 있다. 이에 드론의 안정적인 비행을 지원하는 관제, 통신, 그리고 네트워크 기술에 대한 연구가 끊임없이 진행되고 있다^[1].

드론의 비행 방식은 크게 두 가지로 나뉘는데 지상 제어 센터에서 직접 비행을 제어하는 수동 제어 방식과 미리 입력된 데이터와 인식기능을 이용하는 자율 비행 방식이 있다. 현재 개발되거나 상용화된 드론은 대부분 수동 제어 방식을 따르며 드론 한 대와 이를 제어하는 지상 제어 센터로 구성되어 있다. 그러나 최근에는 군집 드론과 같이 다수의 드론을 그룹으로 묶어 비행 임무를 수행하는 Flying Ad Hoc Network (FANET)에 대한 연구가 진행되면서, 다수의 드론을 효율적으로 제어할 수 있는 자율 비행 방식에 대한 연구가 진행되고 있다.

FANET 환경에서 드론을 제어하기 위해 다양한 연구들이 진행되었다. [2]에서는 드론 간 데이터를 효율적으로 전달하기 위한 최적의 라우팅 알고리즘과 군집 드론의 경로 계획 방법을 제안하였다. 그러나 드론의 빠른 이동성과 급변하는 주변 환경으로 인해 드론들 사이의 링크 단절이 발생할 수 있고 이로 인해 드론의 분실 및 추락과 같은 사고의 위험이 있어 현실적으로 적용하기 어려운 한계가 있다. 이에 [3]에서는 FANET 환경에서 다수의 드론들 간 지속적인 통신을 통해 안정적인 링크를 유지할 수 있다고 주장하였다. 좀 더 구체적으로, 드론들이 FANET 구조로 다중홉을 거쳐 지상 제어 센터와 연결되면 예측 불가능한 상황에서도 보다 신뢰성 있는 링크를 지원할 수 있다는 주장이다. 그리고 [4]에서는 이러한 FANET 구조에 대한 개념을 정립하였고, 쓰임새를 정리하였다. 그러나 링크 단절이나 FANET 복구에 대한 구체적인 해결책은 다루지 않았다.

앞서 살펴보았듯이, 안정적인 드론 제어를 위해서는 링크 단절과 네트워크 복구에 대한 연구가 필요하다. [5]에서는 단일 드론이 사용자의 시야에서 사라질 경우, "리턴 투 홈" 기술을 통해 드론의 비행 시작 지점으로 복귀하는 연구를 진행하였다. 드론의 복귀는

GPS 위치 정보를 활용하여 이루어지며 GPS 측위 오차 극복을 위해 셀 분할 알고리즘과 GPS/INS에 확장 칼만 필터를 적용한 방법을 사용하였다. [6]에서는 단일 드론이 조종 범위를 벗어나게 되었을 때, 이를 스스로 인지하고 조종 가능한 범위로 다시 돌아오는 자동 회귀 드론에 대한 연구가 진행되었다. 자동 회귀 드론은 수행된 이동 명령어의 종류, 명령어 수행 시점의 배터리 잔량을 기억하여 조종 범위를 벗어났을 때 이를 역으로 수행하는 방식이다. 이 방안은 GPS 없이 데이터베이스를 이용한 백트래킹 알고리즘에 초점을 맞추었으며, 퍼지 제어를 통한 보상값으로 신호를 증폭시켜서 오차를 줄이는 방법을 사용하였다. 그러나 [5], [6]은 단일 드론에 중점을 두었다는 한계가 있고, 최근 트렌드인 다중 드론 기반의 FANET 환경에 적용하기에는 한계가 있다. 이에 [7]에서는 다중 드론 기반의 FANET 환경에서 실제의 드론 운용을 고려한 네트워크 자가 복구 기술이 제안되었다. 이는 각각의 드론이 자신의 IP주소와 GPS 위치 정보가 포함된 Hello 메시지를 주기적으로 브로드캐스트를 통해 이웃 테이블을 생성, 유지, 관리 하고 그 정보의 유지 시간인 TTL(Time-to-live)이 만료가 되면 이웃 테이블의 다른 엔트리를 살펴서 다른 이웃 드론과 연결이 끊어져 있는 경우 단절로 판단한 후에 3단계에 걸친 복구를 수행하게 된다. 그러나 여러 드론의 무리가 지상 제어 센터의 제어로부터 벗어나게 되면 단절을 인지 못한다는 문제점이 있다.

본 논문에서는 다중 드론 기반의 FANET 환경에서 링크 단절을 신속하게 인지하고 네트워크를 스스로 복구 할 수 있는 네트워크 자가복구 기술을 제안한다. 제안하는 기술은 드론 간의 동적인 애드혹 네트워크 환경에서 홉 수(hop count)를 이용하여 단절 의심 단계를 거쳐서 단절을 판단하고 네트워크에서 이탈된 드론이 스스로 네트워크 복구 과정을 수행한다. 시뮬레이션 결과를 통해 제안 기법의 효과성을 입증하였다.

본 논문은 다음과 같이 구성된다. 2장에서는 다중 드론 기반의 FANET 환경에서 제안하는 네트워크 자가 복구 기술에 대해 설명하고, 3장에서는 제안하는 기술에 대한 MATLAB 시뮬레이션 환경을 설명하고 성능을 평가 분석하며, 4장에서 결론을 맺는다.

II. 본 론

제안하는 기법은 다수의 드론들이 임무를 수행하는 FANET 환경에서 링크 단절을 신속하게 체크하기 위한 수단으로서 지상 제어 센터로부터의 홉 수를 이용한다. 각 드론은 이웃 드론들과 정보를 주고받으며 이웃 테이블 정보를 유지하는데, 이 때, 테이블에 지상 제어 센터로부터의 홉 수도 저장하여 드론 스스로 링크 단절을 신속하게 인지할 수 있도록 관리하고 있다.

이웃 테이블은 이웃 드론의 ID 정보, 정보의 유지 시간을 나타내는 TTL 정보, 홉 수 정보로 구성된다. 이웃 드론의 ID 정보 항목은 새로운 이웃인지의 여부를 판별하고 이웃 테이블을 관리하기 위해 사용된다. TTL 값은 Hello 메시지의 전송 주기의 3배로 설정되며, 이웃 테이블의 엔트리 유지 관리를 위해 사용된다. 마지막으로 홉 수는 단절 인지 및 운영환경 통제를 위해서 사용된다.

그림 1은 각각의 드론의 홉 수를 나타낸 것이다. (a)에서 두 드론 사이에 연결이 끊어진 경우 홉 수가 4인 드론은 주변에 더 이상 자신보다 낮은 홉 수가 없어서 단절이라고 판단할 수 있다. 하지만 (b)와 같이 실제로는 단절이 아니게 된다. 이러한 상황을 위해 단절에 대한 판단의 정확성을 높이기 위해 단절 의심 절차를 수행하였다.

드론 스스로 단절이 의심되는 경우, 자신의 홉 수를 hc_{∞} (모든 드론의 수보다 큰 값으로 설정)로 바꾸고 이웃 테이블의 엔트리를 모두 삭제하며, 최대 단절 의심 시간(T_D) 동안 단절을 판단한다. 이 때, 단절로부터의 복구를 위해 드론은 움직이지 않고 제자리에서 단절 여부를 판단한다. 최대 단절 의심 시간 동안에 단절이 아님을 판단하지 못하면 단절로 판단하고 복구동작을 수행하게 된다.

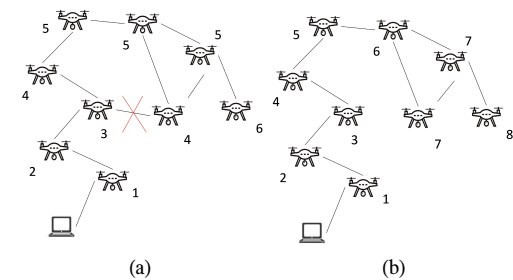


그림 1. 단절이 아닌 상황
Fig. 1. Non-disconnected situation

2.1 메시지 수신

지상 제어 센터로부터의 홉 수가 hc_{∞} 가 아닌 드론은 정상적으로 임무를 수행중인 드론이고, 홉 수가 hc_{∞} 인 드론은 단절 의심 단계로서 단절을 판단하거나 단절 단계로서 복구중인 드론이다. 그리고 각 드론이 관리하는 이웃 테이블 엔트리에 해당되는 드론들은 모두 정상적으로 임무를 수행중인 드론들로 홉 수가 hc_{∞} 가 아닌 드론들에 해당한다.

그림 2는 메시지 수신시의 플로우차트이다. 자신의 홉 수 A와 수신 받은 메시지의 홉 수 B에 따라 크게 4가지 경우로 나뉘게 된다.

첫 번째로 자신의 홉 수가 hc_{∞} 인 드론이 홉 수가 hc_{∞} 인 드론으로부터 메시지를 수신 받은 경우 엔트리를 갱신하지 않는다.

두 번째로 자신의 홉 수가 hc_{∞} 인 드론이 홉 수가 hc_{∞} 가 아닌 드론으로부터 메시지를 수신 받은 경우 엔트리를 생성 하고 자신의 홉 수를 갱신한다.

세 번째로 자신의 홉 수가 hc_{∞} 가 아닌 드론이 홉 수가 hc_{∞} 인 드론으로부터 메시지를 수신 받은 경우 ID정보를 이용하여 이웃테이블을 관리한다. 더 이상 이웃 테이블에 엔트리가 존재 하지 않는 경우는 자신이 네트워크에서 이탈된 상태이므로 단절 단계에 들어간다. 반면 이웃 테이블에 엔트리가 존재하는 경우 네트워크로부터 연결을 확인하기 위해 자신의 이웃테

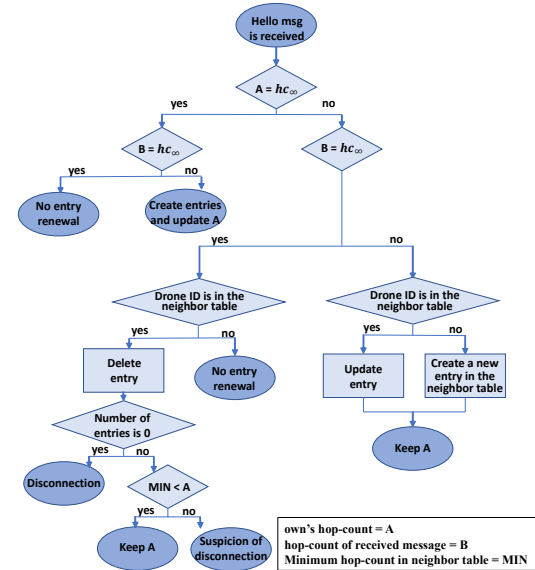


그림 2. 메시지 수신시의 플로우차트
Fig. 2. Flowchart for receiving a message

이블에서 자신의 홉 수보다 작은 홉 수가 존재하는지 확인한다. 존재하지 않는 경우 단절의심 단계를 실행하게 된다.

마지막으로 자신의 홉 수가 hc_{∞} 가 아닌 드론이 홉 수가 hc_{∞} 가 아닌 드론으로부터 메시지를 수신 받은 경우 해당 드론의 ID가 이웃 테이블에 있는지 확인하고 있다면 해당 엔트리를 갱신해주고 없다면 이웃 테이블에 새 엔트리를 생성한 후 자신의 홉 수를 갱신한다.

2.2 TTL 만료

TTL 값이 Hello 메시지 전송주기의 3배로 설정되었기 때문에, 3번의 Hello 메시지를 전달받지 못하면 해당 드론에 대한 TTL 값이 만료되게 된다. 이웃 테이블의 해당 엔트리에 TTL이 만료되었을 때의 절차는 그림 3과 같다. 먼저 이웃 테이블의 해당 엔트리를 삭제한다. 더 이상 이웃 테이블에 엔트리가 존재하지 않는 경우는 자신이 네트워크에서 이탈된 상태이므로 단절 단계에 들어간다. 반면 이웃 테이블에 엔트리가 존재하는 경우 네트워크로부터 연결을 확인하기 위해 자신의 이웃테이블에서 자신의 홉 수보다 작은 홉 수가 존재하는지 확인한다. 존재하지 않는 경우 단절의심 단계를 실행하게 된다.

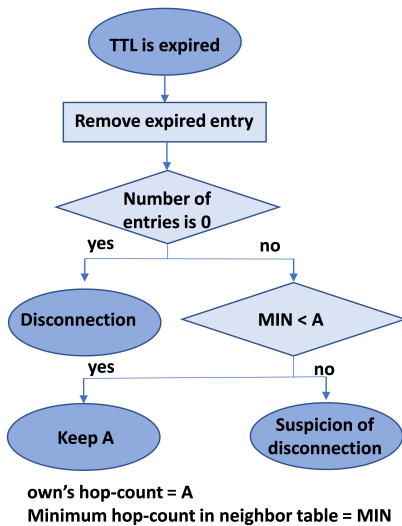


그림 3. TTL 만료 시 플로우차트
Fig. 3. Flow Chart for TTL Expiration

2.3 단절 의심 단계

단절 의심 단계에서는 자신의 홉 수를 hc_{∞} 으로 갱신한 후 메시지 브로드캐스트하고, 최대 단절 의심

시간동안 단절을 판단한다. 홉 수가 hc_{∞} 가 아닌 메시지를 수신하면 홉 수를 갱신하고 임무를 계속 수행해나간다. 반면에 최대 단절 의심 시간동안 단절 의심 드론의 홉 수가 갱신되지 않는다면 단절로 판단하고 단절 단계를 시작한다.

그림 4는 최대 단절 의심 시간을 나타낸 그림이다. 최대 단절 의심 시간 T_U 는 불안정한 홉 수 정보를 해결하고자 메시지를 수신하지 않는 시간 T_U 와 메시지를 수신하는 시간으로 구성된다.

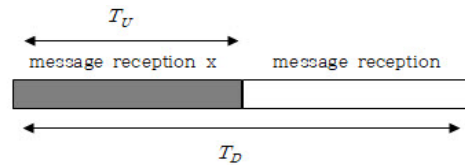


그림 4. 최대 단절 의심 시간
Fig. 4. Maximum Disconnection suspicion Time

2.4 단절 단계

단절 시 드론은 지상 제어 센터의 위치로 이동하며 이동 중에 홉 수가 hc_{∞} 가 아닌 메시지를 수신하면 엔트리 생성 및 자신의 홉 수를 갱신한다. 이는 지상 제어 센터로부터 직접 혹은 다중 홉으로 연결되었음을 의미하므로 복구를 멈추고 임무를 수행한다.

III. 실험

이번 장에서는 제안하는 기법의 성능 평가를 위해 [7]에서 제안한 다중 드론 단계적 네트워크 복구기술을 기존 기법으로서 비교한다. 좀 더 구체적으로, 기존 기법은 홉 수를 기반으로 하진 않기 때문에 hc_{max} (최대 홉 수)가 노드 수와 같은 경우 제안 기법과 기존 기법의 비교를 통하여 성능 평가를 진행하였다.

또한, 제안기법의 자체적인 성능 평가를 위해서는 단절 의심 파라미터에 따른 비교와 hc_{max} 에 따른 비교를 진행하였다. 홉 수가 hc_{max} 이상인 경우 단절로 판단을 하게 된다.

모의실험은 MATLAB을 통해 이루어졌으며, 성능 분석에 사용된 파라미터들의 값은 표 1에 명시되어있다. 지상 제어 센터에서 출발하는 모든 드론은 3차원 환경으로 최고 속도를 정해두고 랜덤 이동을 하는 모델로 가정하였다. 드론의 최대 이동 반경은 3축으로 각각 20Km로 제한을 두었으며, 드론 간의 간격 또는 지상 제어 센터와 드론의 간격이 50m인 경우 Hello

표 1. 성능 분석의 인자와 값
Table 1. Parameters and Values of performance analysis

Parameters	Values
Drone Maximum speed	7.5, 15 m/s
Simulation time	1000 sec
Number of drone	20, 40
Mobility model	Random Waypoint Mobility Model
hc_{max}	1~40
T_D	$(1-6) * T_M$
T_U / T_D	16.66~83.33%
TTL	$3 * T_M$
Transmission range	50m

메시지가 정상적으로 전달이 되는 통신환경이 비교적 좋지 않은 상황을 가정하였다. 모든 결과는 각 모의실험을 100회 반복 수행하여 평균을 도출한 것이다.

3.1 드론 수에 따른 비교

드론의 최대 속도를 15m/s로 매초 랜덤 이동, hc_{max} 는 40, T_D 는 $3 * T_M$, T_U / T_D 는 50%으로 설정하였고, 드론의 수는 각각 20대인 경우와 40대인 경우로 모의실험을 진행했다.

그림 5는 시간에 따른 평균 네트워크 단절시간을 나타낸 것으로 드론의 수에 따라 제안기법과 단계적 복구기법의 비교를 나타내었다. 같은 속도의 환경에서 제안된 기법 단절시 효과적으로 복구가 일어났음을 알 수 있는 반면 기존의 기법은 단절을 인지하지 못하여 효율적으로 복구를 하지 못하였음을 알 수 있다.

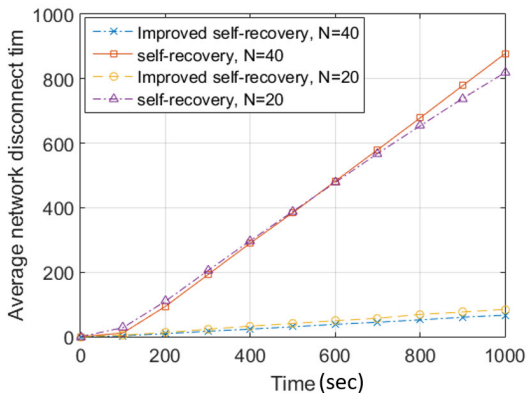


그림 5. 시간에 따른 평균네트워크 단절시간 (드론 수 비교)
Fig. 5. Average network disconnection time according to Time (Comparison of the number of drones)

드론의 수가 더 높은 경우의 평균 네트워크 단절시간이 더 낮은 이유는 총 단절시간은 드론의 수가 많을수록 크겠지만 그 시간을 총 드론의 수로 나눈 수치이며, 드론의 수가 많을수록 하나에 드론의 주변에 더 많은 드론이 존재할 확률이 높아서 낮은 수의 드론에 비해 비교적 지상 제어 센터로부터 다중 홉으로 연결을 잘 유지함을 알 수 있다.

3.2 드론의 속도에 따른 비교

드론의 수는 20대, hc_{max} 는 20, T_D 는 $3 * T_M$, T_U / T_D 는 50%으로 설정하였고, 드론의 최대 속도는 각각 7.5m/s인 경우와 15m/s인 매초 랜덤 이동하는 경우를 가정하여 모의실험을 진행했다.

그림 6은 시간에 따른 평균 네트워크 단절시간을 나타낸 것으로 드론의 속도에 따라 제안기법과 단계적 복구기법의 비교를 나타내었다. 두 기법 모두 속도가 높을수록 평균 네트워크 단절시간이 더 높음을 알 수 있다. 이는 급격하게 드론이 움직일수록 다중 홉에서 이탈하는 드론들이 많음을 의미하고 제안기법이 이러한 환경에서 더욱 효율적임을 확인할 수 있다.

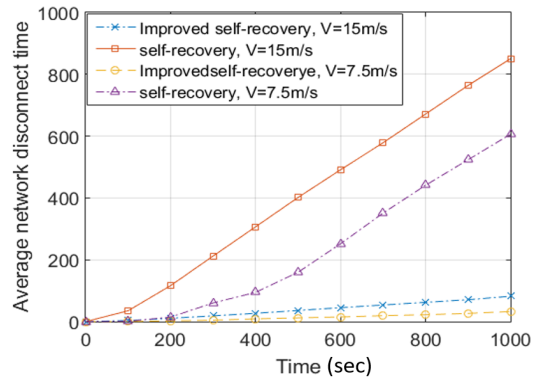


그림 6. 시간에 따른 평균네트워크 단절시간 (드론 속도 비교)
Fig. 6. Average network disconnection time according to Time (Comparison of the speed of drone)

3.3 단절 의심 파라미터 비교

단절 의심 절차에서 임의시간 대기기를 하게 되는데 이때 사용하는 파라미터를 비교하여 효율적인 단절 의심을 하고자 한다. 비교하고자 하는 파라미터는 최대 단절 의심 시간 T_D 와 이 시간 중에서 메시지를 수신하지 않고 대기하는 시간 T_U 의 비율이다. 최대 단절 의심 시간은 무분별하게 길게 잡으면 단절을 판단하는데 많은 시간을 소모하기 때문에 비교적 빠른 판단을 위해 T_D 를 Hello 메시지 전송주기(T_M)의

1~6배로 설정하였고, 메시지를 수신하지 않고 대기하는 시간의 비율(T_U/T_D)을 16.66~83.33%로 설정하였다.

드론의 수는 20대, 최대 15m/s의 속도로 이동하고, hc_{max} 는 20, 드론 간의 간격 또는 지상 제어 센터와 드론의 간격이 50m인 경우 Hello 메시지가 정상적으로 전달이 되는 통신환경으로 설정하였다. 결과는 단절 의심으로부터 판단된 결과 중 정상으로 판단한 비율을 통해 확인했다.

표 2는 단절 의심의 판단결과를 나타낸 것이다. 판단결과는 소수점아래 첫째자리에서 반올림을 하였다. 같은 환경이라면 정상이라고 판단한 비율이 높을수록 효율적으로 단절을 판단했음을 의미한다. 대체적으로 T_U/T_D 가 비슷한 경우에는 T_D 가 클수록 단절 의심으로부터 정상이라고 판단하는 비율이 높아짐을 확인할 수 있었고, T_D 가 같은 경우에는 T_U/T_D 가 낮을수록 단절 의심으로부터 정상이라고 판단하는 비율이 높아짐을 확인할 수 있다. 따라서 최종적으로는 T_D 가 크고 T_U/T_D 가 낮을수록 효율적인 단절 의심이 가능함을 알 수 있다.

단절 의심시간이 길어지면 단절로 판단되는데 걸리는 시간이 길어지게 되고 이는 단절로부터 복구하는데도 시간이 길어짐을 의미하므로 비효율적이다. 따라서 적당히 긴 최대 단절 의심 시간을 설정해야하며, 메시지를 수신하지 않고 대기하는 비율은 낮을수록 좋지만 휴 수가 인 메시지가 주위로 충분히 전달될 시간은 필요하므로, 이를 고려하여 T_D 는 T_M 의 5~6배, T_U/T_D 는 16~25%가 적절하다.

3.4 hc_{max} 에 따른 비교

드론의 수는 20대로 설정하였고, 최대 15m/s의 속도로 움직이고, T_D 는 $6 * T_M$, T_U/T_D 는 33.33%이며, hc_{max} 를 1~20으로 변경하며 진행하였다. 드론은

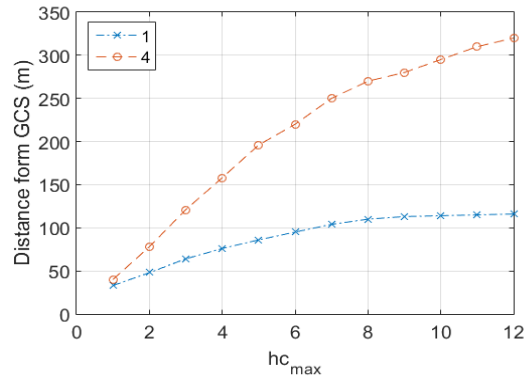


그림 7. hc_{max} 에 따른 평균 활동영역 비교
Fig. 7. Comparison of average activity area according to hc_{max}

각각 1초와 4초간 랜덤 이동을 유지하게 하였다. 드론 간 간격 또는 지상 제어 센터와 드론 사이의 간격이 50m인 경우 Hello 메시지가 정상적으로 전달이 되는 통신환경으로 설정하였다.

그림 7은 hc_{max} 에 따른 평균 활동 영역을 나타낸 것이다. 평균 활동영역은 지상 제어 센터로부터 모든 드론의 거리를 합한 뒤 평균으로 구한 것이다. 랜덤 이동 환경이 다른 둘을 비교해보면 매초 랜덤 이동을 하는 경우는 hc_{max} 가 증가할수록 드론의 활동영역이 커지다가 비슷한 수준이 유지됨을 알 수 있다. 반면에 4초를 주기성으로 랜덤 이동을 하는 모델은 hc_{max} 에 따른 드론의 활동영역이 크게 차이가 난다는 것을 알 수 있다. 따라서 드론이 랜덤하게 이동하는 경우 hc_{max} 을 제한하는 방법으로 드론의 임무 수행반경을 제한할 수 있음을 알 수 있다.

그림 8은 4초간 랜덤 이동을 유지하는 모델로 hc_{max} 에 따른 단절시간과, 단절 의심시간, 단절로부터 회복된 횟수를 비교한 것이다.

단절 의심 시간의 경우는 활동영역과 비슷한 경향

표 2. 단절 의심 판단결과
Table 2. disconnection suspicion result

T_U/T_D T_D	16.66	20	25	33.33	40	50	60	66.66	75	80	83.33
$T_M * 6$	88	-	-	86	-	84	-	80	-	-	75
$T_M * 5$	-	86	-	-	83	-	81	-	-	75	-
$T_M * 4$	-	-	83	-	-	78	-	-	74	-	-
$T_M * 3$	-	-	-	-	79	-	-	73	-	-	-
$T_M * 2$	-	-	-	-	-	73	-	-	-	-	-

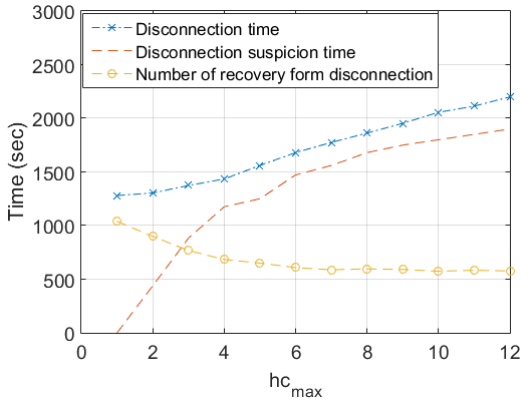


그림 8. hc_{max} 에 따른 파라미터 비교
 Fig. 8. Comparison of parameters according to hc_{max}

을 보인다. hc_{max} 따라 증가한다. 단절시간은 또한 hc_{max} 가 증가함에 따라 증가하는 경향을 보이며 단절로부터 회복횟수를 비교해보면 hc_{max} 낮을 때는 hc_{max} 가 증가함에 따라 감소하다가 hc_{max} 가 일정 수준에 도달하면 비슷한 값을 가지게 된다. 이는 hc_{max} 가 낮을수록 hc_{max} 의 제한으로 인한 단절이 보다 많기 때문에 전체적인 단절횟수가 많아지게 되고 단절로부터 회복되는 횟수 또한 많아지게 되는 것이다. 또한 단절시간과 단절로부터의 회복횟수를 보면 hc_{max} 가 낮을수록 단절로부터 더욱 빠른 회복을 함을 알 수 있다.

결론적으로 hc_{max} 가 낮을수록 활동반경이 작고 단절은 많이 일어나지만 단절로부터 회복이 빠름을 알 수 있고, hc_{max} 가 높아지면 활동반경, 단절시간이 길어짐을 알 수 있다.

IV. 결 론

기존 연구에서는 이웃 테이블을 통해 드론 고립 혹은 네트워크 단절을 인지하고, 이웃 드론의 이전 위치 정보를 기반으로 자율 비행을 수행하는 자율적 네트워크 복구 알고리즘을 제안한 바 있다. 그러나 다수의 드론의 경우, 지상 제어 센터의 제어에서 벗어날 경우 각각의 드론들은 서로가 이웃 테이블의 엔트리에 존재하기 때문에 지상 제어 센터로부터의 단절을 인지하지 못하는 문제점이 있었다. 본 논문에서는 이 문제점을 해결하는 기법으로서 지상 제어 센터로부터의 홉 수를 이용하는 자가복구 알고리즘을 제안하였다. 이를 기존 논문과 비교하여 우수성을 입증하였고, 보

다 정확한 단절인지를 위한 단절 의심의 파라미터를 비교를 통하여 높은 성공률을 가지는 단절 의심 파라미터를 찾았다. 또한 홉 수를 통해 운용환경을 통제할 수 있음을 모의실험을 통하여 검증하였다.

References

- [1] H. Baek and J. Lim, "Design of future UAV-Relay tactical data link for reliable UAV control and situational awareness," *IEEE Commun. Mag.*, vol. 56, no. 10, pp. 144-150, Oct. 2018.
- [2] M. H. Tareque, M. S. Hossain, and M. Atiquzzaman, "On the routing in flying ad hoc networks," in *FedCSIS*, pp. 1-5, Lodz, Poland, Sep. 2015.
- [3] E. W. Frew and T. X. Brown, "Networking Issues for Small Unmanned Aircraft Systems," *J. Intell. Robot. Syst.*, vol. 54, no. 1, pp. 21-37, Jan. 2008.
- [4] W. Zafar and B. M. Khan, "Flying ad-hoc networks: Technological and social implications," *IEEE Technol. Soc. Mag.*, vol. 35, no. 2, pp. 67-74, Jun. 2016.
- [5] S. H. Choi and Y. K. Kim, "Outdoor precision position estimation system using multiple GPS and EKF," *J. Korea Robotics Soc.*, vol. 8, no. 2, pp. 129-135, 2013.
- [6] K. M. Sun, S. J. Jang, D. S. Kwon, and J. S. Ahn, "Self-retraceable drone," in *Proc. KICS Fall Conf.*, pp. 463-465, Nov. 2015.
- [7] G. Kim and Y. Cho, "A stepwise network self-recovery scheme for multi-drone ad hoc networks," *J. KICS*, vol. 43, no. 1, pp. 110-120, Jan. 2018.

김 동 혁 (Donghyeok Kim)



2019년 2월 : 아주대학교 국방
디지털융합학과 학사
<관심분야> 전자공학, 통신공
학, 광통신 공학

백 호 기 (Hoki Baek)



2006년 2월 : 아주대학교 정보
및 컴퓨터공학 학사
2008년 2월 : 아주대학교 정보
통신공학 석사
2014년 2월 : 아주대학교 컴퓨
터공학 박사
2014년 3월~2015년 2월 : 아주

대학교 장위국방연구소 선임연구원
2015년 3월~현재 : 아주대학교 국방디지털융합학과
특임교수
<관심분야> 전술데이터링크, 다중 접속, 시간동기,
위치인식, 무선네트워크, 무인기 통신 네트워크
[ORCID:0000-0001-9213-7845]

임 재 성 (Jaesung Lim)



1983년 2월 : 아주대학교 전자
공학 학사
1985년 2월 : KAIST 영상통신
석사
1994년 2월 : KAIST 디지털통
신 박사
1998년 3월~현재 : 아주대학교
소프트웨어융합학과 정교수

2004년 3월~현재 : 아주대학교 국방전술네트워크 연
구센터장
<관심분야> 이동 및 위성통신, 무선네트워크, 국방
전술통신
[ORCID:0000-0003-0080-9398]