

사이버물리시스템 환경에서 지연마감시간 기반 군집위성 분산 실시간 스케줄링 기법

박 정 원¹ · 이 순 승² · 박 상 혁^{3*}

¹공군교육사령부 항공우주통제학교 우주감시체계정비교관

²공군교육사령부 항공우주통제학교 우주작전교관

³공군교육사령부 항공우주통제학교 교관

Postponed Deadline-Based Distributed Real-Time Scheduling for Satellite Constellations in Cyber-Physical Systems

Jeong-Won Park¹ · Soon-Seung Lee² · Sang-Hyuk Park^{3*}

¹SSSM Instructor, ROKAF Aerospace School, Jinju 52634, Korea

²SO Instructor, ROKAF Aerospace School, Jinju 52634, Korea

³Professor, ROKAF Aerospace School, Jinju 52634, Korea

[요 약]

본 논문은 모자이크 전쟁 환경에서 CPS 기반 군집위성의 분산 실시간 스케줄링 문제를 다룬다. CPS 환경에서는 사이버요소 뿐만 아니라 이동성, 위치, 접근시간 등 물리적 요소를 함께 고려해야 하나 기존 단일 노드 기법은 이를 충분히 반영하지 못한다. 특히 군집위성에서는 특정 위성에서 실패한 임무라도 다른 위성으로 재할당 시 수행 가능하다. 이를 위해 지연마감시간(Postponed Deadline)을 도입하고 군집 전체 관점에서 임무 가능성을 판단하는 PD-DRTS를 제안한다. 제안 기법은 접근시간, 자세기동시간, 수행시간을 반영한 유효슬랙(effective slack) 기반으로 임무를 동적으로 할당하며, 위성과 표적이 모두 이동하는 이중 이동 환경을 고려하였다. STK 시뮬레이션 결과, PD-DRTS는 마감시간 준수율 약 42%, 평균 임무 성공률 약 66%를 달성하였으며, Random, FIFO, LST 기법이 동일 평가 지표에서 약 27~28% 수준에 머무른 것과 비교하여 높은 성능을 보였다.

[Abstract]

This study addresses the distributed real-time scheduling problem for satellite constellations in CPS environments under mosaic warfare, where flexible cooperation and real-time decision-making are crucial. CPS-based scheduling must consider computational resources and physical factors such as mobility, location, and access time, which conventional single-node approaches capture inadequately. In satellite constellations, tasks that miss deadlines on one satellite can still be completed through reassignment. To address this, we introduce a postponed deadline and propose a distributed real-time scheduling algorithm, PD-DRTS, which evaluates mission feasibility from a constellation-wide perspective. The proposed method computes effective slack using access time, attitude maneuvering time, and execution time, and dynamically assigns tasks to the most suitable satellite. A double mobility environment, where both satellites and targets move, is also considered. STK-based simulation results show that PD-DRTS achieves approximately 42% deadline satisfaction and 66% average mission success, exhibiting a performance higher than Random, FIFO, and LST algorithms, which remain at approximately 27–28% for the same metrics.

색인어 : 지연마감시간, 모자이크전, 군집위성, 분산실시간 시스템, 스케줄링

Keyword : Postponed Deadline, Mosaic Warfare, Satellite Constellations, Distributed Real-Time Systems, Scheduling

<http://dx.doi.org/10.9728/dcs.2026.27.6.1637>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 13 April 2026; Revised 08 May 2026

Accepted 04 June 2026

*Corresponding Author; Sang-Hyuk Park

Tel: +82-55-750-5493

E-mail: ggang705@nate.com

1. 서 론

최근 전장 환경은 네트워크 중심전(Network-Centric Warfare, NCW)을 넘어, 다양한 이기종 자산들이 분산적으로 협력하여 임무를 수행하는 모자이크 전쟁(Mosaic Warfare) 개념으로 발전하고 있다[1]-[3]. 특히 이란-미국 간 중동 분쟁과 러시아-우크라이나 전쟁 사례에서는 드론, 위성, 센서 등 다양한 자산들이 실시간으로 연동되어 운용되는 양상이 두드러지게 나타나고 있다[4],[5]. 이러한 변화는 분산된 자산 간 협업과 신속한 의사결정의 중요성을 더욱 부각시키고 있다. 이러한 전장 환경은 다양한 물리적 요소와 계산 시스템이 결합된 사이버물리시스템(Cyber-Physical Systems, 이하 CPS)의 특성과 밀접하게 연관된다[6],[7].

CPS 환경에서는 작업 수행 시 계산 자원뿐만 아니라 이동성, 위치, 접근 가능 시간과 같은 물리적 요소를 함께 고려해야 하며, 시간 제약 하에서 작업을 수행하는 실시간성이 핵심적인 특징으로 작용한다[8]. 국방, 항공, 교통, 로봇, 에너지 시스템 등 다양한 분야에서 실시간 스케줄링은 시스템의 안정성과 성능을 결정짓는 핵심 요소로 활용되고 있다[9],[10]. 기존의 실시간 스케줄링 기법들은 주로 CPU, 네트워크, 메모리와 같은 사이버 자원을 중심으로 설계되어 왔으며, FIFO(First-In First-Out), EDF(Earliest Deadline First), LST(Least Slack Time First) 등의 알고리즘이 대표적으로 활용되어 왔다[11]. 그러나 CPS 환경에서는 작업 수행 과정에서 물리적 이동이 수반되며, 이에 따른 이동 지연과 위치 변화가 실시간 성능에 직접적인 영향을 미친다[8]. 따라서 기존의 사이버 중심 스케줄링 기법을 그대로 적용할 경우 실제 CPS 환경의 특성을 충분히 반영하기 어렵다는 한계가 존재한다[6],[7]. 이러한 한계를 극복하기 위해 저자의 기존 연구 및 특허에서는 이동시간을 고려한 유효 슬랙(effective slack) 기반 스케줄링 기법과 지연마감시간(Postponed Deadline) 개념이 제안되었다[12],[13]. 해당 접근은 특정 노드에서 마감시간을 만족하지 못하는 작업이라 하더라도, 시스템 내 다른 노드로 재할당할 경우 수행 가능성이 존재함을 보여주었다[7]. 이는 협업을 통해 기존에 실패로 판단되던 작업을 성공 가능한 작업으로 재해석할 수 있음을 의미한다. 따라서 본 논문에서 다루는 지연마감시간 개념은 완전히 새로운 개념의 제안이 아니라, 기존 CPS 기반 분산 실시간 스케줄링 연구에서 제안된 개념을 군집위성 환경으로 확장하여 적용한 것이다. 그러나 기존 연구들은 서비스 노드는 고정되어 있고 서비스 대상만 이동하는 환경을 가정하고 있어, 보다 복잡한 물리적 상호작용을 충분히 반영하지 못하는 한계를 가진다[7]. 특히 이러한 접근은 위성과 같이 서비스 노드 자체가 지속적으로 이동하는 우주 환경에는 직접적으로 적용하기 어렵다. 최근에는 다수의 위성이 협력하여 임무를 수행하는 군집위성(constellation) 환경이 주목받고 있다[14],[15]. 군집위성 시스템에서는 위성 자체가 궤도 운동에 따라 지속적으로

이동하며, 동시에 관측 대상 또한 시간에 따라 위치가 변화한다. 즉, 서비스 노드와 서비스 대상 노드가 모두 이동하는 이중 이동 환경(double mobility environment)이 형성된다. 이러한 환경에서는 단일 노드 또는 단일 이동만을 고려한 기존 스케줄링 모델로는 실제 시스템을 효과적으로 반영하기 어렵다. 특히 군집위성 환경에서는 특정 위성 기준으로는 마감시간을 만족할 수 없는 임무라 하더라도, 다른 위성으로 재할당할 경우 마감시간 내 수행이 가능한 경우가 발생한다. 이는 지연마감시간 개념이 군집위성 환경에서도 효과적으로 적용될 수 있음을 시사한다. 이에 본 연구에서는 CPS 환경에서 제안된 지연마감시간 개념을 군집위성 시스템으로 확장하고, 서비스 노드와 서비스 대상이 모두 이동하는 이중 이동 환경을 고려한 분산 실시간 스케줄링 기법을 제안한다. 제안 기법은 각 임무에 대해 위성별 접근시간(access time), 자세기동시간(slew time), 임무 수행시간(execution time)을 종합적으로 고려하여 예상 완료 시각과 유효슬랙(effective slack)을 계산한다. 이후 개별 위성 기준의 수행 가능성만을 판단하는 것이 아니라, 군집 내 모든 위성의 유효슬랙(effective slack)을 비교하여 군집 수준의 유효슬랙(constellation-level slack)을 산출하고, 이를 기반으로 임무 수행 가능성을 판단한다. 또한 특정 위성에서 마감시간을 만족하지 못하는 임무라도 후속 위성 또는 다른 위성의 접근 가능 시간 내에서 수행 가능한 경우에는 협업 기반 재할당(collaborative reassignment)을 수행하도록 하여, 단일 위성 기준에서 실패로 판단될 수 있는 임무를 군집 전체 관점에서 수행 가능한 임무로 재해석한다. 이를 통해 단일 위성 기준으로 수행이 어려운 임무를 협업 기반 재할당을 통해 수행 가능하도록 전환함으로써, 전체 시스템의 마감시간 준수율과 임무 성공률을 향상시키고자 한다.

본 연구에서는 제안 기법의 성능을 검증하기 위해 STK(System Tool Kit) 기반의 군집위성 시뮬레이션 환경을 구성하였다. 시뮬레이션에서는 동일 궤도를 갖는 저궤도위성(Low Earth Orbit, LEO) 군집을 대상으로 위성 간 시간적 비동기성, 표적 접근 윈도우, 자세기동 제약, 임무 수행시간을 반영하였다. 이를 통해 제안 기법이 기존의 Random, FIFO, LST 기반 스케줄링 기법과 비교하여 군집위성 환경에서 마감시간 준수율과 임무 성공률을 향상시킬 수 있는지를 분석하였다. 본 논문의 주요 기여는 다음과 같다. 첫째, CPS 환경에서 제안된 지연마감시간 개념을 군집위성 환경으로 확장하였다. 둘째, 서비스 노드와 서비스 대상이 모두 이동하는 이중 이동 환경을 고려한 실시간 스케줄링 모델을 제시하였다. 셋째, 군집위성 환경에서 협업 기반 임무 재할당이 가능한 분산 실시간 스케줄링 기법(PD-DRTS)을 제안하였다. 넷째, 제안 기법을 통해 기존 단일 위성 기준으로 마감시간을 만족하지 못하는 임무를 협업을 통해 수행 가능하도록 전환할 수 있음을 보였다.

II. 관련 연구 및 한계 분석

실시간 스케줄링은 다양한 시스템에서 핵심적인 연구 주제로 다루어져 왔으며[10],[16],[17], 특히 CPS 환경에서는 시간 제약을 만족시키기 위한 다양한 접근 방법이 제안되어 왔다[7],[8]. 본 장에서는 본 연구와 관련된 기존 연구를 전통적 실시간 스케줄링, CPS 기반 실시간 스케줄링, 위성 임무 스케줄링, 군집위성 협업 스케줄링의 관점에서 분석한다. 또한 각 연구 분야의 한계를 정리하고, 본 연구에서 제안하는 지연마감시간 기반 군집위성 분산 실시간 스케줄링 기법과의 차별성을 명확히 한다.

2-1 전통적 실시간 스케줄링

전통적인 실시간 스케줄링 기법으로는 FIFO(First-In First-Out), EDF(Earliest Deadline First), RM(Rate Monotonic), LST(Least Slack Time First) 등이 대표적으로 사용되어 왔다[1],[10]. 이러한 기법들은 주로 CPU 및 네트워크와 같은 사이버 자원의 효율적인 활용을 통해 작업의 마감시간을 만족시키는 것을 목표로 한다. 특히 EDF와 LST는 마감시간과 유효슬랙을 고려한 동적 우선순위 기반 알고리즘으로 실시간 시스템에서 널리 활용되고 있다. 그러나 이러한 기법들은 작업 수행 과정에서 발생하는 물리적 이동, 위치 변화, 접근 시간 등의 요소를 고려하지 않기 때문에, CPS와 같이 물리적 특성이 중요한 환경에서는 적용에 한계를 가진다[8]. 또한 전통적 실시간 스케줄링 기법은 대부분 단일 노드 또는 개별 자원 기준으로 작업의 수행 가능성을 판단하므로, 다수의 이동 노드가 협업하는 군집위성 환경에서 임무 수행 가능성이 위성 간 재할당에 의해 변화하는 상황을 직접적으로 반영하기 어렵다.

2-2 CPS 기반 실시간 스케줄링

CPS 환경에서는 물리적 이동, 위치 변화, 센서 및 액추에이터 간 상호작용이 시스템 성능에 직접적인 영향을 미친다. 이에 따라 기존의 사이버 중심 스케줄링 기법을 확장하여 물리적 요소를 함께 고려하는 연구가 활발히 진행되어 왔다[8]. 저자의 기존 연구에서는 서비스 노드가 작업 위치로 이동해야 하는 상황을 고려하여, 이동시간을 포함한 유효슬랙(effective slack) 기반 스케줄링 기법과 지연마감시간(Postponed Deadline) 개념이 제안되었다[12],[13]. 해당 접근은 특정 노드에서 마감시간을 만족하지 못하는 작업이라 하더라도, 다른 노드로 재할당될 경우 수행 가능성이 존재함을 보였다[7]. 이는 협업을 통해 기존에 실패로 판단되던 작업을 성공 가능한 작업으로 재해석할 수 있음을 의미한다. 다만 기존 CPS 기반 지연마감시간 연구는 제한된 이동 환경을 대상으로 하며, 서비스 노드와 서비스 대상이 모두 시간에 따

라 이동하는 이중 이동 환경(double mobility environment)을 충분히 고려하지 못한다. 특히 군집위성 환경에서는 위성 자체가 궤도 운동에 따라 지속적으로 이동하고, 표적 또는 관측 대상 또한 시간에 따라 위치가 변화하므로, 기존 CPS 스케줄링 모델을 그대로 적용하기 어렵다. 따라서 기존 CPS 기반 지연마감시간 개념을 군집위성 환경에 적용하기 위해서는 위성별 접근시간, 자세기동시간, 임무 수행시간을 함께 고려한 새로운 수행 가능성 판단 구조가 필요하다.

2-3 위성 임무 스케줄링

위성 임무 스케줄링은 제한된 시간과 자원 내에서 다양한 관측 요청을 효율적으로 처리하기 위한 문제로, 오랫동안 연구되어 온 분야이다. 일반적으로 위성 임무 스케줄링에서는 표적의 관측 가능 시간, 위성의 궤도 및 위치 변화, 자세기동 제약, 임무 수행 시간, 에너지 및 저장 용량 제약 등이 주요 고려 요소로 작용한다[18]. 기존 연구들은 주로 최적화 기반 기법이나 휴리스틱 알고리즘을 활용하여 임무 수행 수 또는 총 이득을 최대화하는 방향으로 진행되어 왔다[19],[20]. 그러나 이러한 접근 방식은 대부분 오프라인 또는 준실시간 환경을 가정하며, 엄격한 실시간 마감시간 제약을 직접적으로 고려하지 않는 경우가 많다. 또한 다수의 위성이 존재하는 환경에서도 개별 위성 단위로 문제를 분리하여 해결하는 경우가 일반적이며, 군집 전체 관점에서의 협업 기반 스케줄링은 상대적으로 제한적으로 연구되어 왔다. 즉, 기존 위성 임무 스케줄링 연구는 관측 기회와 자원 제약을 고려한다는 점에서 현실적인 장점을 가지지만, 단일 위성 기준으로 마감시간을 만족하지 못하는 임무가 다른 위성의 후속 접근을 통해 수행 가능해지는 현상을 실시간 마감시간 충족 가능성(deadline feasibility) 관점에서 해석하는 데에는 한계가 있다.

2-4 군집위성 협업 스케줄링

최근에는 다수의 위성이 협력하여 임무를 수행하는 군집위성 환경이 주목받고 있으며, 이를 위한 협업 기반 스케줄링 기법들이 제안되고 있다[14],[15]. 이러한 연구들은 임무 분산, 자원 공유, 통신 협력을 통해 전체 시스템 성능을 향상시키는 것을 목표로 한다. 그러나 기존 군집위성 스케줄링 연구들은 다음과 같은 한계를 가진다. 첫째, 임무 할당은 수행되지만 실시간 마감시간 기반의 동적 재할당 문제를 충분히 고려하지 않는다. 둘째, 협업에 따른 임무 수행 가능성 변화를 마감시간 개념의 확장으로 해석하지 않는다. 셋째, 서비스 노드와 서비스 대상이 동시에 이동하는 이중 이동 환경에 대한 명확한 모델링이 부족하다. 특히 군집위성 환경에서는 각 위성의 접근 가능 시간과 표적 위치가 시간에 따라 변화하므로, 동일 임무라 하더라도 어느 위성에 할당되는지와 어느 시점에 수행되는지에 따라 유효슬랙 값이 달라진다. 따라서 군집 위성 협업 스케줄링에서는 단순한 임무 분산뿐만 아니라, 군

집 전체 관점에서 임무 수행 가능성을 판단하는 군집위성 수준 슬랙(constellation-level slack) 기반 접근이 필요하다.

최근 대규모 군집위성 및 지구관측위성 스케줄링 문제는 분산 제약 최적화(Distributed Constraint Optimization Problem, DCOP)와 분해 기반 분산 스케줄링 관점에서도 연구되고 있다[14],[21]. 이러한 연구들은 대규모 위성군에서 관측 요청을 효율적으로 할당하고 전체 스케줄 품질을 향상시키는 데 기여하였다. 그러나 대부분 관측 이득, 요청 만족도 또는 전체 스케줄 품질 향상에 초점을 두며, 본 연구와 같이 실시간 마감시간 제약과 군집 수준 유효슬랙을 이용하여 임무 수행 가능성을 즉시 판단하고 재할당하는 문제는 제한적으로 다룬다.

2-5 기존 연구의 한계 및 본 연구의 차별성

종합하면, 전통적 실시간 스케줄링 기법은 물리적 이동을 고려하지 않으며, CPS 기반 스케줄링 기법은 이중 이동 환경을 충분히 반영하지 못한다. 또한 위성 임무 스케줄링 연구는 실시간 마감시간 제약을 직접적으로 다루지 않는 경우가 많으며, 군집위성 스케줄링 연구는 협업은 고려하나 마감시간 확장 개념에 대한 연구는 제한적이다. 특히 기존 연구에서는 단일 노드 기준으로 마감시간을 만족하지 못하는 임무를 협업을 통해 재해석하는 개념이 부족하며, 서비스 노드와 서비스 대상이 모두 이동하는 환경을 동시에 고려한 실시간 스케줄링 모델은 충분히 다루어지지 않았다. 그리고, 기존 작업 이전(task migration) 기반 접근은 주로 부하 분산(load balancing)이나 자원 활용률 향상을 목적으로 하며, 작업을 다른 노드로 이동시켜 시스템 전체의 처리 효율을 높이는 데 초점을 둔다. 반면 본 연구에서 다루는 군집위성 환경에서는 단순히 작업을 다른 노드로 이동시키는 것만으로는 충분하지 않다. 위성과 표적이 모두 이동하기 때문에, 동일 임무의 수행 가능성은 시간, 위성 위치, 접근 가능 시간, 자세기동시간에 따라 동적으로 변화한다. 따라서 본 연구는 기존 task migration과 달리, 후속 위성의 접근 가능성과 위성 간 시간적 비동기성을 활용하여 마감시간 충족 가능성(deadline feasibility) 자체를 군집 수준에서 재정의한다. 또한, 기존 유효슬랙 기반 스케줄링은 일반적으로 개별 노드의 지역 유효슬랙(local slack)을 기준으로 작업의 우선순위를 결정한다. 그러나 PD-DRTS는 군집 내 모든 위성에 대해 유효슬랙을 계산하고, 그중 가능한 유효슬랙 값을 이용하여 군집위성 수준 슬랙(constellation-level slack)을 정의한다. 이를 통해 단일 위성 기준에서는 마감시간을 만족하지 못하는 임무라도, 군집 내 다른 위성에 의해 수행 가능한 경우에는 실패 임무가 아니라 재할당 가능한 임무로 판단할 수 있다.

이에 본 연구에서는 CPS 환경에서 제안된 지연마감시간 개념을 우주 기반 군집위성 환경으로 확장하고, 이중 이동 환경을 고려한 분산 실시간 스케줄링 기법(PD-DRTS)을 제안한다. 제안 기법은 협업 기반 임무 재할당을 통해 마감시간을

만족하지 못하는 임무를 수행 가능한 임무로 전환함으로써, 군집 전체의 스케줄링 성능을 향상시키는 것을 목표로 한다.

III. 군집위성 기반 CPS 시스템 모델 및 문제 정의

3-1 CPS 기반 군집위성 환경

사이버물리시스템(CPS)은 사이버 시스템과 물리 시스템이 긴밀하게 결합된 구조로, 시간 제약 하에서 물리적 작업을 수행하는 특성을 가진다[6],[8]. CPS 기반 실시간 스케줄링에서는 계산 자원뿐만 아니라 작업 수행 위치, 이동성, 접근 가능 시간, 물리적 제약이 함께 고려되어야 하며, 이러한 요소들은 작업의 완료 시각과 마감시간 만족 여부에 직접적인 영향을 미친다[7]. 기존 CPS 스케줄링 연구에서는 서비스 노드가 고정되고 서비스 대상만 이동하는 단일 이동 환경(single mobility environment)을 중심으로 실시간 스케줄링 문제가 다루어져 왔다. 그러나 실제 군집위성 환경에서는 위성 자체가 궤도 운동에 의해 지속적으로 이동하며, 동시에 관측 대상 또한 시간에 따라 위치가 변화한다. 즉, 서비스 노드와 서비스 대상이 모두 이동하는 이중 이동 환경(double mobility environment)이 형성된다. 이러한 환경에서는 접근 시간, 자세기동 시간, 표적의 위치 변화가 동시에 고려되어야 하며, 이는 기존 CPS 기반 스케줄링 모델보다 훨씬 복잡한 실시간 의사결정 문제를 유발한다.

본 연구에서는 이러한 군집위성 기반 CPS 환경을 이중 이동 환경으로 정의하고, 위성별 접근시간(access time), 자세기동시간(slew time), 임무 수행시간(execution time)을 반영하여 임무 수행 가능성을 판단한다. 특히 단일 위성 기준에서 마감시간을 만족하지 못하는 임무라도, 군집 내 다른 위성의 후속 접근을 통해 마감시간 내 수행 가능한 경우가 존재하므로, 임무 수행 가능성을 개별 위성이 아닌 군집 전체 관점에서 평가한다.

3-2 표적 및 위성 모델

군집위성 시스템에서 표적 집합과 위성 집합을 각각 다음과 같이 정의한다.

$$T = \{ 1, 2, \dots, N \}, S = \{ 1, 2, \dots, K \}$$

각 표적 $i \in T$ 다음과 같은 속성을 가진다.

d_i : 마감시간 (deadline),

e_i : 임무 수행 시간(execution time)

각 위성 $k \in S$ 는 시간 t 에 따라 위치 및 자세 상태가 변화

하며, 표적 i 를 수행하기 위해 다음과 같은 시간이 필요하다.

$a_i^{(k)}$: 표적 i 에 대한 접근 시간,
 $s_i^{(k)}$: 자세기동 시간(slew time)

따라서 위성 k 가 시각 t 에서 표적 i 를 수행할 경우 예상 완료 시각은 다음과 같이 정의된다.

$$f_i^{(k)} = t + a_i^{(k)} + s_i^{(k)} + e_i \quad (1)$$

이는 위성이 표적에 접근하고, 자세를 정렬한 후 임무를 수행하는 전체 소요 시간을 반영한 완료 시각을 의미한다.

3-3 단일 위성 환경에서의 수행 가능성

단일 위성 환경에서 표적 i 가 마감시간을 만족하기 위한 조건은 다음과 같다.

$$f_i^{(k)}(t) \leq d_i \quad (2)$$

이를 유효슬랙(effective slack)으로 정의하면 다음과 같다.

$$l_i^{(k)}(t) = d_i - f_i^{(k)}(t) \quad (3)$$

따라서,

$$l_i^{(k)}(t) \geq 0: \text{수행 가능}, \\ l_i^{(k)}(t) < 0: \text{수행 불가능}$$

단일 위성 환경에서는 임무 수행 가능 여부가 해당 위성 k 의 상태에 의해 결정되며, 수행 불가능한 임무는 즉시 실패로 간주된다.

3-4 다중 위성 환경에서의 수행 가능성

다중 위성 환경에서는 하나의 표적 i 가 군집 내 여러 위성 중 하나에 의해 수행될 수 있다. 위성 집합을 $S = \{1, 2, \dots, K\}$ 로 정의할 때, K 는 전체 위성의 수를 의미하며, k 는 개별 위성을 나타내는 인덱스이다. 시각 t 에서 위성 k 가 표적 i 를 수행할 경우의 완료 시각 $f_i^{(k)}(t)$ 에 대해, 해당 임무가 마감 시간을 만족하기 위한 조건은 다음과 같다.

$$\exists k \in S \quad \text{s.t.} \quad f_i^{(k)}(t) \leq d_i \quad (4)$$

각 위성에 대한 유효슬랙(effective slack)은 다음과 같이 정의된다.

$$l_i^{(k)}(t) = d_i - f_i^{(k)}(t) \quad (5)$$

군집 전체 관점에서의 유효슬랙(effective slack)은 다음과 같이 정의한다.

$$l_i^{\text{constellation}}(t) = \max_{k \in S} l_i^{(k)}(t) \quad (6)$$

식 (6)은 시각 t 에서 표적 i 에 대해 군집 내 모든 위성 중 가장 큰 유효슬랙(effective slack) 값을 선택함으로써, 군집 전체 관점에서의 수행 가능성을 나타낸다. 따라서 다음과 같은 판단 기준을 얻을 수 있다.

$$l_i^{\text{constellation}}(t) \geq 0: \text{수행 가능}, \\ l_i^{\text{constellation}}(t) < 0: \text{수행 불가능}$$

이는 임무 수행 가능성이 개별 위성이 아닌 군집 전체의 상태에 의해 결정됨을 의미한다.

3-5 지연마감시간(Postponed Deadline)의 정의

단일 위성 환경에서는 다음과 같은 경우 임무가 실패로 간주된다.

$$l_i^k(t) < 0$$

그러나 다중 위성 환경에서는 다음과 같은 상황이 발생할 수 있다.

$$l_i^k(t) < 0, \exists K' \in S \quad \text{s.t.} \quad l_i^{(K')}(t) \geq 0 \quad (7)$$

즉, 특정 위성에서는 수행이 불가능하더라도 다른 위성에서는 수행이 가능한 경우가 존재한다. 이를 군집 관점에서 재정의하면 다음과 같다.

$$l_i^{\text{constellation}}(t) \geq 0 \quad \text{and} \quad \min_{k \in S} l_i^{(k)}(t) < 0 \quad (8)$$

본 연구에서는 이러한 현상을 지연마감시간(Postponed Deadline)으로 정의한다. 즉, 지연마감시간이란 단일 위성 기준으로 마감시간을 만족하지 못하는 임무가 군집 내 다른 위성으로 재할당됨으로써 결과적으로 마감시간을 만족하게 되는 현상을 의미한다.

3-6 수식 기반 분석의 한계와 시뮬레이션 기반 접근

군집위성 환경에서는 다음과 같은 특성으로 인해 수식 기반 분석에 한계가 존재한다. 첫째, 위성 수와 표적 수가 증가

함에 따라 가능한 임무 할당 조합이 조합적으로 증가한다. 둘째, 접근 시간과 자세기동 시간은 시간에 따라 동적으로 변화하며, 표적 또한 이동한다. 셋째, 특정 임무 선택이 이후 임무의 수행 가능성에 영향을 미치는 상호 의존성이 존재한다. 이러한 특성으로 인해 전체 시스템을 폐쇄형 수식으로 분석하는 것은 어렵다. 따라서 본 연구에서는 단일 위성 및 소규모 환경에서는 수식 기반 분석을 적용하고, 다중 위성 및 복잡한 환경에서는 시뮬레이션 기반 접근을 적용한다. 특히 위성의 궤도 운동, 접근 시간, 자세기동 등을 현실적으로 반영하기 위해 STK(System Tool Kit)를 활용한 시뮬레이션 환경을 구성한다. 또한 실제 군집위성 운용에서는 통신 지연, 에너지 제약, 위성 간 협업 비용 등이 임무 재할당 성공률에 영향을 줄 수 있다. 본 연구의 기본 모델에서는 접근시간, 자세기동시간, 임무 수행시간을 중심으로 유효슬랙(effective slack)을 계산하며, 이 중 자세기동시간과 자세기동 제약은 실제 위성 운용에서 에너지 소모 및 운용 가능성과 밀접하게 관련되는 요소이다. 따라서 본 연구에서는 별도의 배터리 잔량 모델을 사용하지는 않았으나, 자세기동 제한과 자세기동시간을 통해 에너지 관련 운용 제약을 간접적으로 반영한다.

또한 임무 재할당은 각 위성의 접근 윈도우와 위성별 상태 갱신 이후의 수행 가능성에 따라 제한되도록 구성한다. 따라서 본 연구는 임무 재할당이 각 위성의 접근 윈도우와 위성별 상태 갱신 조건에 의해 제한되는 상황을 고려한다. 다만 통신 지연 시간과 잔여 에너지를 독립 변수로 설정하여 임무 재할당 성공률 변화를 정량적으로 분석하는 민감도 분석은 본 논문의 범위를 넘어서는 내용이므로, 향후 연구에서 통신 지연, 에너지 제약, 위성 간 협업 비용을 포함한 확장 모델로 분석할 예정이다.

본 장에서는 CPS 기반 군집위성 환경을 이중 이동 환경으로 확장하여 정의하고, 단일 위성과 다중 위성 환경에서의 수행 가능 조건을 수식적으로 도출하였다. 이를 통해 지연마감시간 개념이 군집위성 환경에서 자연스럽게 확장될 수 있음을 보였으며, 시뮬레이션 기반 분석의 필요성을 제시하였다.

IV. 제안 기법: PD-DRTS(Postponed Deadline-based Distributed Real-Time Scheduling)

본 장에서는 3장에서 정의한 지연마감시간(Postponed Deadline) 개념을 기반으로, 군집위성 환경에서의 분산 실시간 임무 스케줄링 기법인 PD-DRTS(Postponed Deadline-based Distributed Real-Time Scheduling)를 제안한다. 기존 실시간 스케줄링 기법은 개별 노드 단위에서 임무 수행 가능 여부를 판단하는 반면, PD-DRTS는 군집 전체 관점에서 수행 가능성을 평가하고 협업 기반으로 임무를 재할당하는 것을 특징으로 한다. PD-DRTS는 군집위성 스케줄링 문

제의 조합적 복잡성과 실시간 의사결정 요구를 고려한 휴리스틱(heuristic) 기반 탐욕적 온라인 스케줄링 기법(greedy online scheduling method)이다. 대규모 군집위성 스케줄링 문제에서는 위성 수, 표적 수, 접근 윈도우, 자세기동 제약이 증가함에 따라 가능한 임무 할당 조합이 급격히 증가한다. 최근 분산 최적화 기반 연구에서는 이러한 문제를 해결하기 위해 DCOP 기반 스케줄링 및 분해 기반 분산 스케줄링 기법이 제안되고 있다[14],[21]. 그러나 이러한 접근은 전체 스케줄 품질 향상에는 장점이 있으나, 반복 탐색이나 메시지 교환 과정으로 인해 즉각적인 의사결정이 요구되는 실시간 전장 환경에서는 계산 지연이 발생할 수 있다. 따라서 본 연구의 PD-DRTS는 전역 최적해 보장보다는 실시간 의사결정 가능성에 초점을 두고, 군집 수준 유효슬랙을 기준으로 현재 시점에서 수행 가능한 임무를 즉시 선택하는 탐욕적 온라인 휴리스틱 구조로 설계하였다. 본 연구의 목적은 전체 스케줄링 공간에 대한 전역 최적해(global optimum)를 보장하는 것이 아니라, 매 시점에서 접근시간, 자세기동시간, 임무 수행시간을 반영하여 수행 가능한 임무를 실시간으로 선택하고 할당하는데 있다. 따라서 제안 기법은 실시간 환경에서 계산 복잡도를 낮추면서 마감시간 위반 가능성이 높은 임무를 우선적으로 처리하는 방향으로 설계되었다. PD-DRTS는 3장에서 정의한 예상 완료 시각과 유효슬랙(effective slack)을 기반으로 동작한다. 각 표적 i 와 위성 k 에 대해 다음과 같이 정의한다.

$$\begin{aligned} f_i^{(k)}(t) &= t + a_i^{(k)}(t) + s_i^{(k)}(t) + e_i, \\ l_i^{(k)}(t) &= d_i - f_i^{(k)}(t) \end{aligned} \quad (9)$$

이를 바탕으로 군집 전체 관점의 slack은 다음과 같이 계산된다.

$$l_i^{constellation}(t) = \max_{k \in S} l_i^{(k)}(t) \quad (10)$$

$l_i^{constellation}(t) \geq 0$ 인 경우 해당 임무는 군집 협업을 통해 수행 가능한 것으로 판단한다. 위 연산은 PD-DRTS의 의사결정의 기준이 된다. 여기서 $l_i^{constellation}(t)$ 은 개별 위성의 local slack이 아니라, 군집 내 모든 위성의 수행 가능성을 비교하여 얻은 군집 수준의 유효슬랙이다. 따라서 특정 위성에서는 마감시간을 만족하지 못하는 임무라도, 다른 위성이 충분한 slack을 제공하는 경우에는 재할당 가능한 임무로 판단된다. 이는 기존 작업이전(task migration)이 주로 부하 분산을 목적으로 하는 것과 달리, PD-DRTS가 마감시간 충족 가능성(deadline feasibility) 자체를 군집 수준에서 재평가한다는 점에서 차별성을 가진다.

4-1 PD-DRTS 알고리즘

PD-DRTS는 크게 다음 세 단계로 구성된다.

1) 군집 기반 수행 가능성 평가

각 표적에 대해 모든 위성 $k \in S$ 에 대한 $l_i^{(k)}(t)$ 를 계산하고, 식 (10)을 이용하여 군집 기준 $l_i^{constellation}(t)$ 를 산출한다.

2) 임무 선택

군집 기준 slack이 0 이상인 표적들 중에서 $l_i^{constellation}(t)$ 가 최소인 표적을 선택한다. 이는 마감시간에 가장 압박한 임무를 우선적으로 처리하기 위함이다.

3) 위성 선택 및 할당

선택된 표적 i 에 대해 $l_i^{(k)}(t)$ 가 최대가 되는 위성 k 를 선택하고 해당 임무를 할당한다. 이와 같은 구조를 통해 PD-DRTS는 임무의 긴급성과 자원 활용 효율성을 동시에 고려한다. 다음은 제안 기법의 전체 절차를 나타낸 의사코드이다.

```

Algorithm : PD-DRTS scheduling procedure
Input:
  Satellite set S, Target set T, Current time t
Output:
  Distributed mission schedule
Step 1. Initialize the set of unscheduled targets.
Step 2. For each unscheduled target, compute the completion
time for all satellites considering access time, slew time, and
execution time.
Step 3. Compute the effective slack of each target for each
satellite.
Step 4. For each target, determine the constellation-level slack
by selecting the maximum slack among all satellites.
Step 5. Select only the targets whose constellation-level slack is
non-negative.
Step 6. Among the feasible targets, select the target with the
minimum slack.
Step 7. Assign the selected target to the satellite that provides the
maximum slack.
Step 8. Update the state of the selected satellite and remove the
target from the unscheduled target set.
Step 9. Repeat Steps 2-8 until no schedulable target remains.
    
```

4-2 기존 기법과의 비교

제안 기법은 기존 스케줄링 기법과 비교하여 다음과 같은 차별성을 가진다.

FIFO / Random: 임무 순서 기반 → 협업 고려 없음

LST: 단일 위성 기준 slack → 군집 효과 반영 불가

PD-DRTS: 군집 기준 slack 활용 → 협업 기반 수행 가능성 확장. 특히 PD-DRTS는 단일 위성 기준으로 수행이 불가능한 임무를 군집 협업을 통해 수행 가능하도록 전환할 수 있다는 점에서 기존 기법과 근본적인 차별성을 가진다.

본 장에서는 지연마감시간 개념을 기반으로 군집위성 환경에서의 분산 실시간 스케줄링 기법인 PD-DRTS를 제안하였다. 제안 기법은 군집 전체 관점에서 임무 수행 가능성을 평

가하고, 협업 기반의 재할당을 통해 수행 가능 영역을 확장하는 특징을 가진다. 다만 제안 기법은 휴리스틱 기반 탐욕적 온라인 스케줄링 기법이므로, 전체 스케줄링 구간에 대한 전역 최적해를 보장하지는 않는다. 본 연구에서는 실시간 의사결정이 요구되는 군집위성 환경에서 계산 가능성과 실시간성을 확보하는 데 초점을 두었다. 또한 PD-DRTS는 기존 최적화 기반 군집위성 스케줄링 기법과도 차별성을 가진다. 최적화 기반 기법은 전체 스케줄 구간에 대한 높은 해 품질을 목표로 하지만, 반복 탐색 또는 메시지 교환 과정이 필요할 수 있다[14],[21]. 반면 PD-DRTS는 현재 시점에서 각 위성의 접근시간, 자세기동시간, 임무 수행시간을 이용해 유효슬랙을 계산하고, 군집 수준에서 가장 긴급한 임무를 즉시 선택한다. 따라서 전역 최적성은 보장하지 않지만, 계산 구조가 단순하고 온라인 의사결정에 적합하여 실시간성이 중요한 모자이크 전장 환경에 적용하기 용이하다. 최적화 기반 기법과의 정량 비교는 향후 연구에서 다룰 예정이다.

V. 실험 및 성능 평가

5-1 실험 환경

본 장에서는 제안한 PD-DRTS 기법의 성능을 검증하기 위해 군집위성 환경에서 시뮬레이션을 수행하였다. 본 연구에서는 표적을 단일 지점이 아닌 지역 단위로 모델링하고, 각 지역 내에서 복수의 세부 임무를 생성함으로써 보다 현실적인 작전 환경을 반영하였다. 그림 1, 2는 제안기법 개념도로 위성과 표적이 모두 시간에 따라 변화하는 이중 이동 환경에서 접근 가능 시간과 유효슬랙이 달라질 수 있음을 보여준다. 특히 위성1(Sat1)에서 수행하지 못한 임무가 후속 접근 구간을 갖는 위성2(Sat2)에 의해 수행 가능해질 수 있는 구조를 반영한다.

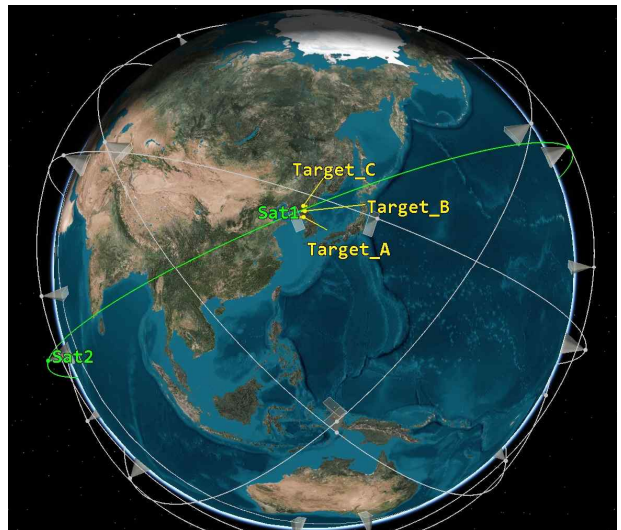


그림 1. STK 시뮬레이션 전체 개념도

Fig. 1. Overall conceptual diagram of the STK simulation

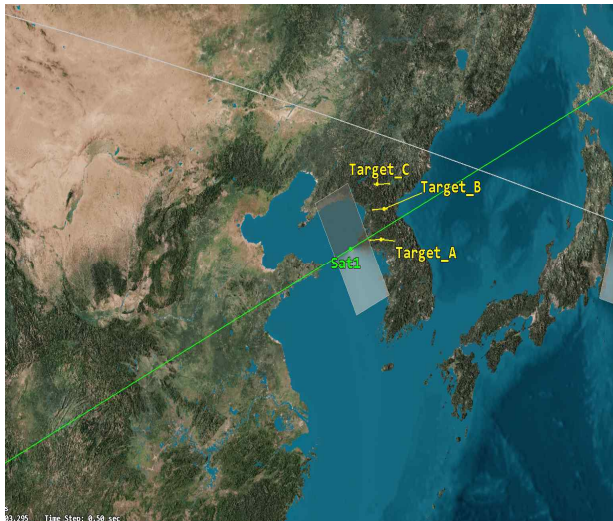


그림 2. STK 시뮬레이션 환경

Fig. 2. STK simulation environment

위성 및 센서 관련 주요 설정은 표 1과 같다. 본 연구에서는 동일 궤도를 갖는 두 개의 LEO 위성을 사용하였으며, 두 위성 간 위상차를 약 20분으로 설정하였다. 이에 따라 동일 표적에 대해 약 20분 간격으로 재접근이 가능하며, 이는 선행 위성에서 수행되지 못한 임무를 후속 위성이 처리할 수 있는 환경을 제공한다. 표 2에서 나타난 바와 같이, 각 위성의 단일 패스 내에서 복수 표적의 접근 구간이 상당 부분 중첩된다. 이는 한반도 작전환경과 유사하여 신뢰성을 형성한다. 또한 Sat2는 Sat1 이후 동일 표적군에 대해 재접근함으로써, 선행

표 1. 군집위성 시뮬레이션 파라미터

Table 1. Satellite constellation simulation parameters

Parameter	Value
Simulation Time	15 Mar 2026, 03:00-04:00 (UTC)
Number of Satellites	2 (LEO constellation)
Orbit	Identical orbit with ~20 min phase difference
Sensor	EO Sensor
Observation Duration	15 sec
Attitude Maneuver Rate	1.5°/sec
Attitude Constraints	Pitch $\pm 10^\circ$, Roll $\pm 25^\circ$

표 2. 지역표적 시뮬레이션 파라미터

Table 2. Regional target simulation parameters

Satellite	Target	Start Time (UTC)	End Time (UTC)	Duration (s)
Sat1	A(A-1,A-2)	03:11:12	03:11:42	29.6
Sat1	B(B-1,B-2)	03:11:23	03:11:52	29.6
Sat1	C(C-1,C-2)	03:11:24	03:11:54	29.9
Sat2	A(A-1,A-2)	03:31:15	03:31:45	29.8
Sat2	B(B-1,B-2)	03:31:24	03:31:54	29.6
Sat2	C(C-1,C-2)	03:31:33	03:32:03	29.8

위성에서 수행되지 못한 임무를 후속 위성에서 처리할 수 있는 구조를 제공한다. 이러한 특성은 본 연구에서 제안한 군집 협업 기반 스케줄링 기법의 효과를 검증하기에 적합하다. 각 표적은 지역 기반으로 모델링되며, 각 지역 내에서 복수의 세부 임무가 생성된다.

임무의 마감시간은 접근 윈도우를 기준으로 설정하였으며, 일부 임무는 선행 위성에서만 수행 가능한 긴급 임무로, 일부는 후속 위성까지 지연이 허용되는 임무로 구성하였다. 또한 다양한 스케줄링 난이도를 반영하기 위해 마감시간을 확률적으로 분포시키고, 단일 위성 환경에서는 수행이 어려우나 군집 협업을 통해 수행 가능해지는 시나리오를 포함하도록 설계하였다.

본 연구에서는 성능 평가를 위해 마감시간 준수율과 평균 임무 성공률을 사용하였다. 마감시간 준수율은 전체 임무 중 마감시간 내에 완료된 임무의 비율을 의미한다. 평균 임무 성공률은 각 지역별로 수행 완료된 임무의 비율을 계산한 후, 이를 모든 지역에 대해 평균한 값으로 정의한다.

5-2 실험결과 및 분석

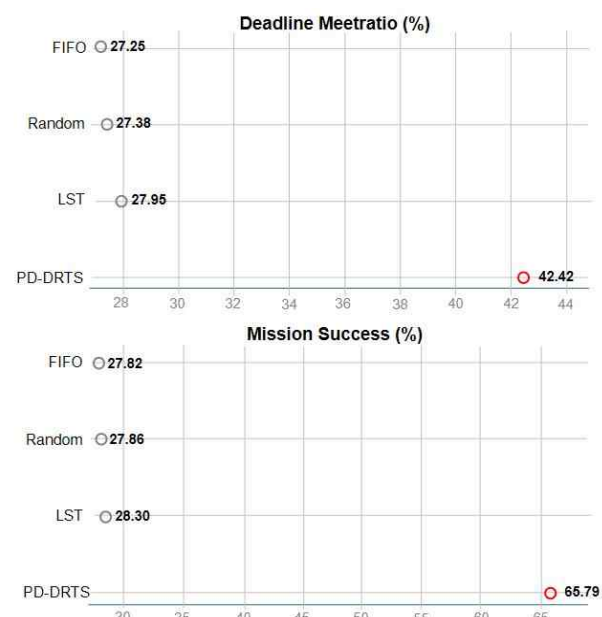


그림 3. 알고리즘별 성능평가 결과

Fig. 3. Performance evaluation results by algorithm

제안된 PD-DRTS 기법의 성능을 기존 기법(Random, FIFO, LST)과 비교하기 위해 마감시간 준수율과 평균 임무 성공률을 기준으로 분석하였다. 그림 3은 알고리즘별 성능 평가 결과를 나타낸다. 실험 결과, PD-DRTS는 마감시간 준수율 약 42.42%를 기록하였으며, Random, FIFO, LST 기법은 약 27~28% 수준에 머무르는 것으로 나타났다. 또한 평균 임무 성공률에서도 PD-DRTS는 약 66% 수준의 성능을 보였으며, 기존 기법들은 약 27~28% 수준에 머물렀다. 이는 PD-DRTS가 개별 위성 기준이 아닌 군집 전체 관점에서 유효슬랙을 계산하고, 가장 적합한 위성에 임무를 동적으로 재

할당함으로써 기존 기법보다 높은 성능을 보였음을 의미한다. 특히 단일 위성 환경에서는 마감시간을 만족하지 못하는 임무라도, 후속 위성 또는 다른 위성을 통해 처리할 수 있기 때문에 마감시간 준수율과 평균 임무 성공률이 모두 향상된 것으로 분석된다. 이러한 결과는 PD-DRTS가 군집 전체 관점에서 slack 기반 수행 가능성을 평가하고, 가장 적합한 위성에서 임무를 동적으로 재할당하는 구조에 기인한다. 특히 이동 환경에서 위성 간 시간적 비동기성을 활용함으로써, 선행 위성에서 수행되지 못한 임무를 후속 위성이 처리할 수 있으며, 이는 지역별 임무 처리율의 균형을 향상시키는 주요 요인으로 작용한다. 종합하면, 제안 기법은 단일 위성 기준의 스케줄링 한계를 극복하고, 군집 협업을 통해 임무 수행 가능 영역을 확장함으로써 마감시간 준수율과 평균 임무 성공률 모두에서 우수한 성능을 달성하였다.

VI. 결론 및 향후연구

본 논문에서는 사이버물리시스템(CPS) 환경에서의 군집 위성 기반 분산 실시간 스케줄링 문제를 다루고, 지연마감시간(Postponed Deadline) 개념을 기반으로 한 PD-DRTS(Postponed Deadline-based Distributed Real-Time Scheduling) 기법을 제안하였다. 제안 기법은 군집 전체 관점에서 임무 수행 가능성을 평가하고, 협업 기반으로 임무를 동적으로 재할당함으로써 기존 단일 노드 중심 스케줄링 기법의 한계를 극복하고자 하였다.

시뮬레이션 결과, PD-DRTS는 마감시간 준수율 약 42%, 평균 임무 성공률 약 66%를 달성하였으며, Random, FIFO, LST 기법이 동일 평가 지표에서 약 27~28% 수준에 머무른 것과 비교하여 높은 성능을 보였다. 이러한 결과는 제안 기법이 군집 수준에서 수행 가능성을 평가하고, 위성 간 협업을 통해 임무를 재배치함으로써 단일 위성 환경에서 수행이 어려운 임무를 효과적으로 처리할 수 있음을 보여준다. 특히 제안 기법은 이동 환경에서 위성 간 시간적 비동기성을 활용하여 임무 재할당을 수행함으로써, 임무 수행 가능 영역을 확장하는 특징을 가진다. 이는 기존 스케줄링 기법으로는 달성하기 어려운 성능 향상을 가능하게 하며, 군집위성 기반 임무 수행의 효율성을 크게 향상시킬 수 있음을 확인했다.

다만 PD-DRTS는 군집위성 스케줄링 문제의 조합적 복잡성과 실시간 의사결정 요구를 고려한 휴리스틱 기반 탐욕적 온라인 스케줄링 기법이므로, 전체 스케줄링 구간에 대한 전역 최적해를 보장하지는 않는다. 본 연구에서는 실시간 환경에서 접근시간, 자세기동시간, 임무 수행시간을 반영하여 수행 가능한 임무를 동적으로 선택하고 할당하는 데 초점을 두었다.

향후 연구에서는 통신 지연, 에너지 제약, 위성 간 협업 비용 등을 고려한 보다 현실적인 스케줄링 모델로 확장하고, 다수의 위성을 포함하는 대규모 군집 환경에서의 성능 검증을 수행할 예정이다. 또한 불확실한 환경에서의 강건한 스케줄링

기법에 대한 연구가 필요하다. 특히 본 연구에서는 자세기동 시간과 자세 제약을 통해 에너지 관련 운용 제약을 간접적으로 반영하였으나, 별도의 배터리 잔량 모델이나 에너지 소비 모델을 명시적으로 포함하지는 않았다. 또한 통신 지연 시간을 독립 변수로 설정한 정량적 민감도 분석은 수행하지 않았으므로, 향후 연구에서는 통신 지연, 잔여 에너지, 위성 간 협업 비용이 임무 재할당 성공률과 마감시간 준수율에 미치는 영향을 정량적으로 분석할 계획이다. 아울러 distributed EDF 및 constellation-level EDF와 같은 변형 기법을 구현하여 PD-DRTS와의 정량 비교를 수행할 예정이다.

감사의 글

이 논문은 2026년 공군사관학교 국고연구비의 지원으로 수행된 연구임(ROKAFA-26-P-1).

참고문헌

- [1] D. S. Alberts, J. J. Garstka, and F. P. Stein, *Network Centric Warfare Developing and Leveraging Information Superiority*, CCRP, 1999.
- [2] S.-H. Park and J.-H. Kim, "Real-Time NCW Systems Using Distributed Processing," *Korea Computer Congress*, Vol. 36, No. 1, pp. 245-249, 2009.
- [3] N. A. O'Donoghue, S. McBirney, and B. Persons, *Distributed Kill Chains: Drawing Insights for Mosaic Warfare from the Immune System and from the Navy*, Santa Monica, CA: RAND Corporation, 2021.
- [4] M. Eslami, "Iran's Drone Supply to Russia and Changing Dynamics of the Ukraine War," *Journal for Peace and Nuclear Disarmament*, Vol. 5, No. 2, pp. 507-518, 2022.
- [5] Z. Momoh and A. L. Malumfashi, "The Strategic Deployment of Unmanned Aerial Vehicles in Contemporary Armed Conflicts: A Comparative Study of the Russia-Ukraine and Israel-Gaza Conflicts," *Kashere Journal of Politics and International Relations*, Vol. 3, No. 4, pp. 202-216, 2025.
- [6] S. Park, J.-H. Kim, and G. Fox, "Effective Real-Time Scheduling Algorithm for Cyber Physical Systems Society," *Future Generation Computer Systems*, 32, pp. 253-259, 2014.
- [7] S.-S. Lee, C.-H. Nam, J.-W. Park, and S.-H. Park, "DR-MWS: Distributed Real-Time Mosaic Warfare System Based on Postponed Deadline," *Journal of Digital Contents Society*, Vol. 24, No. 10, pp. 2337-2343, 2023.
- [8] E. A. Lee, "Cyber Physical Systems: Design Challenges," in *Proceedings of the 11th IEEE International Symposium on*

Object and Component-Oriented Real-Time Distributed Computing, Orlando: FL, pp. 363-369, 2008.

- [9] A. Burns, "Scheduling Hard Real-Time Systems: A Review," *Software Engineering Journal*, Vol. 6, No.3, pp. 116-128, 1991.
- [10] Z. R. M. Azmi, K. A. Bakar, M. S. Shamsir, W. N. W. Manan, and A. H. Abdullah, "Performance Comparison of Priority Rule Scheduling Algorithms Using Different Inter Arrival Time Jobs in Grid Environment," *International Journal of Grid and Distributed Computing*, Vol. 4, No. 3, pp. 61-70, 2011.
- [11] J. W. S. Liu, *Real-Time Systems*, Upper Saddle River, NJ: Prentice Hall, 2000.
- [12] S.-H. Park, J. H. Kim, and B. H. Lee, Real Time Scheduling Method of Cyber Physical System, Patent No. 10-1421603, 2014.
- [13] S. H. Park, J. H. Kim, and K. S. Kim, Real Time Scheduling Method of Cyber Physical System, Patent No. 10-1421601, 2014.
- [14] I. Zilberstein, A. Rao, M. Salis, and S. Chien, "Decentralized, Decomposition-Based Observation Scheduling for a Large-Scale Satellite Constellation," *Journal of Artificial Intelligence Research*, Vol. 82, pp. 169-208, 2025.
- [15] G. Picard, C. Caron, J.-L. Farges, J. Guerra, C. Pralet, and S. Roussel, "Autonomous Agents and Multiagent Systems Challenges in Earth Observation Satellite Constellations," in *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent System*, pp. 39-43, 2021.
- [16] S. H. Park, J. H. Kim, and C. H. Han, "The Extended-Military Multimedia Systems Based on Real-Time Scheduling Scheme," *Journal of the Institute of Electronics Engineers of Korea CI*, Vol. 48, No. 1, pp. 26-32, 2011.
- [17] S. H. Park, J. H. Kim, C. H. Han, and K. S. Kim, "Effective One-to-One Correspondence Method of $O(N^2 \log(N))$ Complexity Between Distributed Units," *International Journal of Advanced Robotic Systems*, Vol. 9, No. 1, pp. 1-7, 2012.
- [18] A. Globus, J. Crawford, J. Lohn, and A. Pryor, "A Comparison of Techniques for Scheduling Earth Observing Satellites," in *Proceedings of the 16th Conference on Innovative Applications of Artificial Intelligence*, San Jose: CA, pp. 836-843, 2004.
- [19] J. C. Pemberton and F. Galiber III, "A Constraint-Based Approach to Satellite Scheduling," in *Constraint Programming and Large Scale Discrete Optimization*, American Mathematical Society, pp. 101-114, 2001.

- [20] D.-H. Cho, J.-H. Kim, H.-L. Choi, and J. Ahn, "Optimization-Based Scheduling Method for Agile Earth-Observing Satellite Constellation," *Journal of Aerospace Information Systems*, Vol. 15, No. 11, pp. 611-626, 2018.
- [21] S. Krigman, T. Grinshpoun, and L. Dery, "Scheduling of Earth Observing Satellites Using Distributed Constraint Optimization," *Journal of Scheduling*, Vol. 27, pp. 507-524, 2024.

박정원(Jeong-Won Park)



2026년 : 경상국립대학교 우주항공공학 (석사)

2017년~2022년: 공군 사이버정보체계운용기사
부서관(ACMI/EWTS, 전자광학위성감시체계
운용)
2023년~현 재: 공군 교육사령부 항공우주통제학교 위성감시
체계정비교관
※관심분야 : 우주영역인식, 우주작전

이순승(Soon-Seung Lee)



2023년 : 서울사이버대학교
기계공학전공 (학사)
2026년 : 경상국립대학교(석사 재학중)

2000년~2022년: 공군 항공통제 부서관(E-737 항공통제사, 레
이더 담당)
2022년~현 재: 공군 교육사령부 항공우주통제학교 전자전교
관 (전문군무경력관 나군)
※관심분야 : 우주영역인식, 우주작전

박상혁(Sang-Hyuk Park)



2013년 : 아주대학교 NCW
(컴퓨터공학 박사)

2000년~2020년: 대한민국 해병대 장교(드론봇 전투체계 등
연구)
2019년~2020년: 아주대학교 유비쿼터스 연구소, 특별연구원
2020년~2021년: 아주대학교 장위국방연구소, 선임연구원
2021년~2022년: 합동참모본부 국방분석실험실 책임연구원
2022년~현 재: 공군 교육사령부 항공우주통제학교 교관(전문
군무경력관)
※관심분야 : 분산실시간시스템, 우주작전, AI