

# 안정성과 실시간성을 고려한 항공전자 시스템 맞춤형 스케줄러의 관한 연구

박상유\*, 최준규, 권도백

엘아이지넥스원(주)

sangyou.park@lignex1.com

## A Study on a Custom Scheduler for Avionics Systems with Safety and Real time Considerations

Park Sang You\*, Choi Jun Kyu, Kwon Do Baeg,

LIGNEX1

### 요 약

본 논문에서는 항공전자 소프트웨어의 자원 비효율성과 동적 환경 대응 한계를 해결하기 위해, 맞춤형 스케줄러(Custom Scheduler)를 제안한다. 기존 시스템은 고정 우선순위(Fixed Priority) 또는 시간 분할(Time Partitioning) 기반 스케줄링으로 안정성을 확보하지만, 자원의 효율적 활용과 실시간 변화 대응에는 한계가 있다. 제안된 스케줄러는 Safety Aware Scheduler (SAS)와 Deadline Diversity Scheduler (DDS)로 구성되며, 각각 DAL 기반 우선순위와 자원 격리, Deadline 인식과 하드웨어 다양성 기반 중복 실행을 통해 안정성과 실시간성을 향상시킨다. 본 연구는 이러한 설계가 기존 정적 스케줄링의 한계를 완화하고, 항공전자 환경에서 효율적이고 안정적인 컨테이너 기반 태스크 배치를 가능하게 함을 보여준다.

### I. 서론

항공전자 소프트웨어는 비행 제어, 센서 융합, 항법, 통신 등 생명과 직결되는 핵심 기능을 수행하기 때문에, 개발 과정 전반에서 높은 수준의 안정성과 추적성이 요구된다. 이러한 요구사항은 DO-178C, ARP4754A, DO-254 등의 국제 표준을 통해 구체화되어 있으며, 각 단계마다 철저한 검증과 문서화 절차를 거쳐야 한다. 그러나 이러한 체계적인 개발 절차는 동시에 몇 가지 문제점을 낳는다. 첫째, 검증된 도구(Validated Tool)에 대한 높은 의존성으로 인해 최신 기술을 유연하게 적용하기 어렵다. 둘째, 복잡한 검증 체계로 인해 시스템 통합 및 변경 시 비용이 과도하게 증가한다. 셋째, 실시간성을 확보하기 위한 하드웨어 종속 구조가 시스템의 확장성과 이식성을 제한한다.

기존 항공전자 시스템은 주로 고정 우선순위(Fixed Priority)나 시간 분할(Time Partitioning) 방식의 스케줄링에 의존한다. 하지만 이러한 정적 방식은 시스템 자원의 효율적 활용이나 동적 환경 변화 대응에는 한계가 있다. 특히, 다양한 미션 프로파일과 실시간 부하 변화가 발생하는 현대 항공전자 환경에서는, 기존 구조만으로 안정성과 효율성을 동시에 확보하기 어렵다.

이에 본 논문은 맞춤형 스케줄러(Custom Scheduler)를 제안한다. 제안된 스케줄러는 안전 등급(DAL) 기반 자원 할당, 태스크 시한(Deadline) 고려 배치, 그리고 하드웨어 다양성(Diversity)[1]을 반영한 중복 실행 전략을 통해 기존 항공전자 소프트웨어 공학이 직면한 비효율 문제를 완화하는 것을 목표로 한다.

### II. 본론

항공전자 시스템의 소프트웨어는 주로 RTOS(Real-Time Operating System) 기반에서 고정 우선순위(Fixed Priority Scheduling) 또는 시간 분할(Time Partitioning) 방식으로 동작한다. 이러한 방식은 일정한 응답 시간과 검증 가능성을 확보하는 데 유리하지만, 시스템 자원의 동적 활용

이나 미션별 요구 변경에 대응하기 어렵다. 특히 정적 자원 할당으로 인해 일시적 부하 변화에 따른 동적 조정이 불가능하며, 검증된 프로세서나 보드에 종속되어 새로운 플랫폼으로의 이식이 어렵다는 하드웨어 의존성 문제가 발생한다. 또한 단순한 코드 수정이라도 전체 검증 프로세스를 다시 거쳐야 하므로, 변경 비용과 시간이 기하급수적으로 증가하게 된다. 이러한 제약으로 인해 항공전자 소프트웨어는 기술적 혁신보다 문서화와 검증 절차의 복잡성에 더 많은 리소스를 소비하게 되며, 결과적으로 실질적 성능 향상보다는 형식적 안정성에 치중하게 된다. 이러한 한계를 해결하기 위해 본 논문에서는 Kubernetes의 기본 스케줄러 구조를 참조하되, 항공전자 시스템 특화 요구사항을 반영한 [2]맞춤형 스케줄러(Custom Scheduler)를 제안한다. 항공전자 환경은 일반 클라우드와 달리 태스크의 실시간성, DAL(Design Assurance Level)에 따른 안전 등급 분리, 하드웨어 다양성 및 이중화 요구 등 특수한 조건을 갖기 때문이다. 제안된 스케줄러는 크게 세 가지 핵심 설계 원칙을 따른다.

첫째, 안전 등급 기반 자원 분리(Safety Level Partitioning)이다. 각 Pod는 DO-178C의 DAL 등급(A~E)에 따라 우선순위를 부여받으며, 스케줄러는 고등급(DAL A~B) 태스크를 저등급(DAL C~E) 태스크와 물리적으로 격리된 노드에 배치하여 안전성 침해 가능성을 원천적으로 차단한다. 둘째, Deadline 인식 스케줄링(Deadline aware Scheduling)이다. EDF(Earliest Deadline First) 알고리즘을 변형 적용하여 태스크의 마감 시한과 현재 클러스터 부하를 동적으로 평가한다. 실시간성 요구가 높은 태스크는 우선 배치되며, 비실시간 태스크는 자원 여유가 있을 때만 배치된다.

셋째, 하드웨어 다양성 기반 이중화(Diversity-based Redundancy Scheduling)이다. 동일 기능을 수행하는 이중 태스크를 서로 다른 아키텍처 노드(예: ARM vs x86)에 분산 실행함으로써 단일 하드웨어 결함이 전체 기능 손실로 이어지지 않도록 한다. 이는 DO-254의 “Dissimilar Redundancy” 개념을 소프트웨어 스케줄링 계층에서 구현하는 것이다.

제안된 맞춤형 스케줄러의 구체적 설계 예시는 두 가지 유형으로 나뉜다 [3].

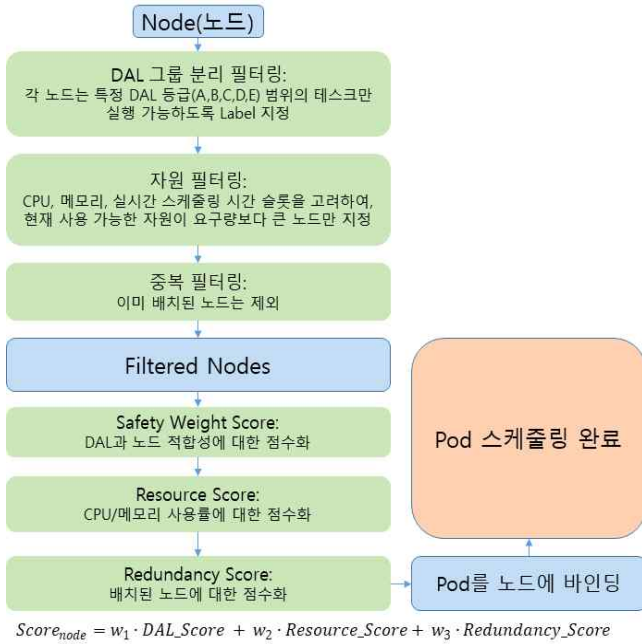


그림 1 Safety Aware Scheduler 흐름도

Safety Aware Scheduler (SAS)는 테스크의 DAL 등급에 따라 노드를 분리하고 각 노드의 자원을 동적으로 재조정한다. SAS는 Multi-level Priority Queue 구조를 활용하며, 상위 큐에는 DAL A~B 수준의 핵심 임무 테스크를, 하위 큐에는 지원용 테스크를 배치한다. 스케줄링 사이클은 Filtering 단계에서 DAL 적합성, 자원 여유, 중복 실행 가능성을 기준으로 후보 노드를 선별하고, Scoring 단계에서는 DAL Weight 순위, 자원 활용 점수, 중복 실행 최적화 점수를 계산하여 최종 배치 노드를 결정한다. 이를 통해 핵심 안전 테스크 간섭을 최소화하고 검증 과정을 단순화하며 시스템 안정성을 향상시킨다.

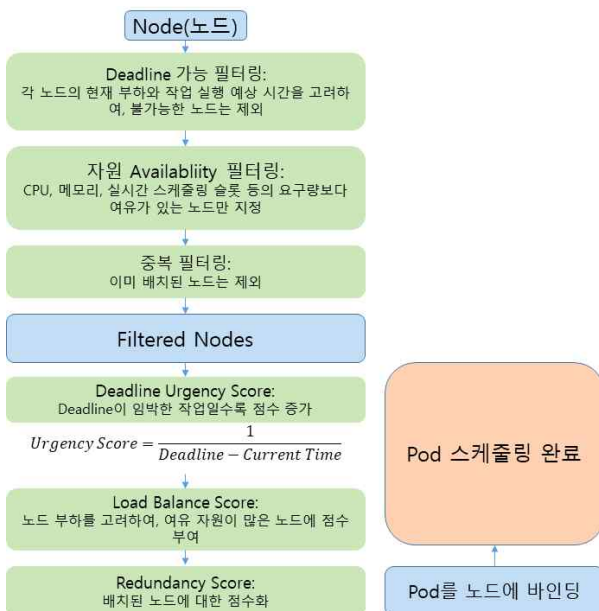


그림 2 Deadline Aware Scheduler 흐름도

Deadline Diversity Scheduler (DDS)는 시한 인식 EDF 알고리즘과 다양성 기반 중복 실행을 결합한 형태이다. DDS는 Filtering 단계에서 각 테스크의 Deadline 가능 여부, 자원 가용성, 중복 실행 가능성을 기준으로 고려하여 후보 노드를 선정하며, Scoring 단계에서는 마감 압박 테스크 우선 배치, 부하 균형, 중복 실행 최적화 점수를 계산하여 최종 배치 노드를 결정한다. 이를 통해 실시간성을 확보하고, 장애 허용성을 높이며, 자원 활용을 최적화할 수 있다. SAS와 DDS는 각각 안전성 중심과 실시간성 중심으로 최적화된 스케줄링 전략을 제공하며, 항공전자 시스템의 다양한 운영 요구를 효과적으로 충족할 수 있다.

### III. 결론

본 논문에서 제안한 맞춤형 스케줄러 구조는 항공전자 환경에서도 컨테이너화된 모듈을 안전하고 결정적으로 배치할 수 있는 현실적인 접근 방안이다. Safety Aware Scheduler는 DAL 기반 우선순위와 자원 격리를 통해 핵심 안전 테스크의 안정성을 확보하고, Deadline Diversity Scheduler는 시한 인식 스케줄링을 고려하여 실시간성 및 장애 허용성을 향상시킨다.

두 스케줄러는 각각 안전성과 실시간성 중심 요구를 충족하면서, 기존 정적 스케줄링 방식의 비효율성을 완화할 수 있는 설계 전략을 제공한다. 향후 연구에서는 제안된 스케줄러 알고리즘을 실제 실시간 커널(RT Linux) 또는 하이퍼바이저 기반 IMA(Integrated Modular Avionics) 환경에 적용하여, 실험적 검증과 인증 절차 간소화 가능성을 분석할 예정이다. 또한 다양한 미션 프로파일과 동적 환경 변화에 대한 성능 평가를 통해, 항공전자 소프트웨어 개발에서 혁신과 안정성을 동시에 확보할 수 있는 추가 최적화 방안을 모색할 계획이다.

### 참 고 문 헌

- [1] Hongtao Wang (2024). Research on Task Scheduling Model of Ant Colony Optimization Cloud Computing Platform for Online Practical Customer-training Application. IEIE Transactions on Smart Processing & Computing, 13(3), 243-253. 10.5573/IEIESPC.2024.13.3.243
- [2] JeongHo Kim, BoAh Choi, YoungEon Yoon, MiJin Kim, Ji-Hoon Bae, & JongHyuk Lee (2021-11-25). Developing a custom scheduler for optimal execution of applications in a Kubernetes environment. Proceedings of KIIT Conference, Jeju.
- [3] Kyungjin Kim, Chaehyun Lim, & Hyunjung Yu (2020-11-13). A Study on Custom Scheduler Design for Users with Procrastination Habits: Focused on mirum Scheduling Kit. KSDS Conference Proceeding, Seoul.