

RAG 기반 3-DOF USV Heading Autopilot 을 위한 설명 가능한 PID 극점 배치 파라미터 자동 튜닝 연구

이승현, 전윤모, 김웅섭*
동국대학교

leesh914@dgu.ac.kr, jumo0716@dongguk.edu, woongsup@dongguk.edu

RAG Based Explainable Automatic Tuning of PID Pole-Placement Parameters for a 3-DOF USV Heading Autopilot

SeungHeon Lee, YoonMo Jeon, Woongsup Kim*
Department of Information Communication Engineering, Dongguk Univ

요약

본 논문은 3-DOF USV(Unmanned Surface Vehicle)의 Heading Autopilot 의 PID 극점 배치 파라미터를 추천하는 검색 중강 생성(RAG) 프레임워크를 제안한다. 오프라인으로 운항 조건- 응답 로그 145,800 건을 BGE-M3 임베딩으로 벡터화해 Chroma DB 에 저장하고, 온라인에서는 BGE-Reranker 로 유사 사례를 선별한 뒤 LLM 에 중강 입력하여 자연 진동수를 산출한다. 제안 시스템 내의 LLM 은 추천 값과 함께 제안 근거 및 이유를 동시에 생성하여 투명성을 확보한다. 본 연구 방식은 근거 중심 추론으로 수동·휴리스틱 의존을 줄이고, 설명 가능한 의사 결정을 구현하며, 다양한 해상 조건에서도 오버슈트·정착 시간·Yaw 응답 등을 안정적으로 제어하는 효율적이고 강건한 튜닝을 달성한다.

I. 서 론

USV(Unmanned Surface Vehicle) Heading Autopilot 성능은 PID 파라미터, 특히 자연 진동수와 감쇠비에 크게 좌우된다. 최근에는 적응형·자체 튜닝 오토파일럿이 보편화되었으나, 다양한 외란 조건에서 최적 파라미터를 선택하기 위해서는 여전히 체계적인 근거가 필요하다. 기존의 수학적 최적화 접근은 복잡한 상호작용을 완전히 모델링하기 어렵고, LLM 단독 활용은 제어공학적 전문성이 부족해 신뢰성에 한계가 있다.

이에 본 연구는 운항 로그 기반 RAG 아키텍처를 적용해 유사 사례를 검색·재정렬하여 LLM 추론을 보강하고, 데이터 기반으로 PID 극점 배치 파라미터를 자동 추천하는 방식을 제안한다. 특히 LLM 이 추천 값뿐 아니라 제안 근거와 이유를 함께 제시하도록 설계하여 결과의 해석 가능성과 실무 적용성을 강화하였다. 이를 통해 수동 튜닝 의존도를 낮추고, 다양한 외란 환경에서도 효율성과 강건성을 동시에 확보할 수 있음을 보인다.

II. 연구 방법론

2.1 시뮬레이션 모델

본 연구에서는 Python 기반 uuv_python_simulator [1]를 활용하여 길이 100m 급 Frigate(호위함)의 3-DOF (Surge, Sway, Yaw) 모델을 구축하였다. 시뮬레이터는 MATLAB MSS Toolbox 를 Python 환경에 맞게 확장한 것으로, 객체 지향적 구조를 통해 다양한 해양 운송체를 모델링할 수 있다. 제어기는

Fossen(2021) [2]에서 제시한 PID 기반 오토파일럿 아키텍처를 적용하였다.

2.2. 로그 데이터베이스 구축

RAG 지식 베이스 구축을 위해 총 145,800 건의 운항 조건별 제어 입력- 응답 로그를 생성하였다. 각 로그에는 입력 변수(조류 속도, 조류 방향, ω_{n_surge} , ω_{n_sway} , ω_{n_yaw}), 출력(X/Y 위치 오버슈트, Yaw 오버슈트, 속도 안정성, 최종 도달 위치), 그리고 Cost Function (1)으로 산출한 성능 점수가 포함된다. 이 성능 점수는 이후 BGE-Reranker 단계에서 후보 로그의 우선순위를 결정하는 핵심 근거로 활용된다. 모든 운항 조건을 포괄하는 것은 비현실적이므로, 본 연구에서는 다양한 해상 환경을 대표하면서도 우수한 성능을 보인 로그를 전략적으로 선별하여 지식 베이스를 구성하였다. 이를 통해 성공 사례 중심의 일반화된 추론을 유도하고, LLM 기반 파라미터 추천의 신뢰도를 높일 수 있다.

$$S = 0.25 \hat{m}_{x_os} + 0.25 \hat{m}_{y_os} + 0.35 \hat{m}_{dist} + 0.10 \hat{m}_{yaw_os} + 0.05 \hat{m}_{vel}. \quad (1)$$

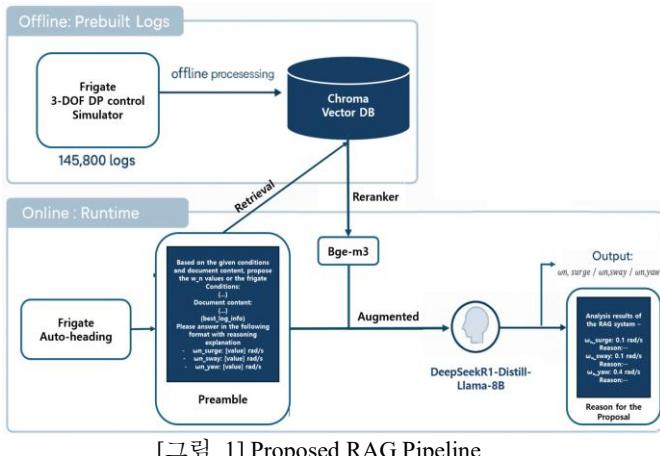
\hat{m}_{x_os} : X 위치 오버슈트 (X Position Overshoot),
 \hat{m}_{y_os} : Y 위치 오버슈트 (Y Position Overshoot),
 \hat{m}_{dist} : 최종 도달 거리 오차 (Distance Error),
 \hat{m}_{yaw_os} : Yaw 오버슈트 (Yaw Overshoot),
 \hat{m}_{vel} : 속도 안정성 (Velocity Stability).

[식 1] Cost Function 정의

2.3 제안된 RAG 파이프라인 구조

오프라인 단계에서는 145,800 개의 시뮬레이션 로그를 BGE-M3 임베딩 모델로 벡터화하여 Chroma DB 에 영구 저장한다. 이 저장소는 런타임과 독립적으로

운영되며, 온라인 추론 시 신속한 근거 검색을 위한 지식 베이스로 활용된다.



온라인 추론 과정은 다음과 같이 단계로 수행된다.

(1) 질의(Query): 사용자는 목표 상태와 해류 조건을 JSON 형식의 Preamble로 입력한다.

(2) 검색(Retrieval): BGE-M3 임베딩을 이용해 Chroma DB에서 관련 로그를 1차 선별한다.

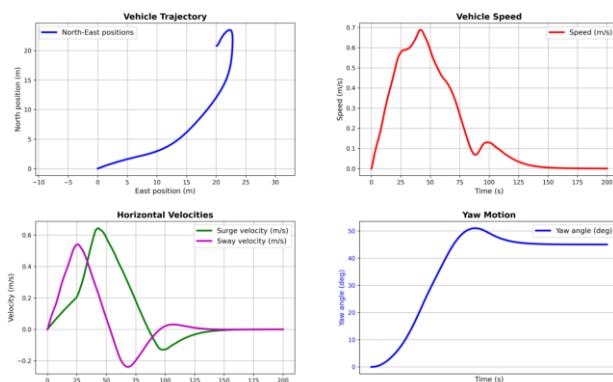
(3) 재정렬(Reranking): BGE-Reranker가 선별된 문서를 교차 인코딩하여 상위 근거(Top-k)를 정밀하게 재정렬한다.

(4) 증강·생성(Augmentation & Generation): 최종 선별된 근거는 Preamble과 결합되어 DeepSeek-R1-Distill-Llama-8B에 입력되며, 모델은 최적의 제어 파라미터(ω_{n_surge} , ω_{n_sway} , ω_{n_yaw})를 추론한다. 결과는 제어 모듈이 즉시 활용할 수 있도록 JSON 형식으로 반환된다.

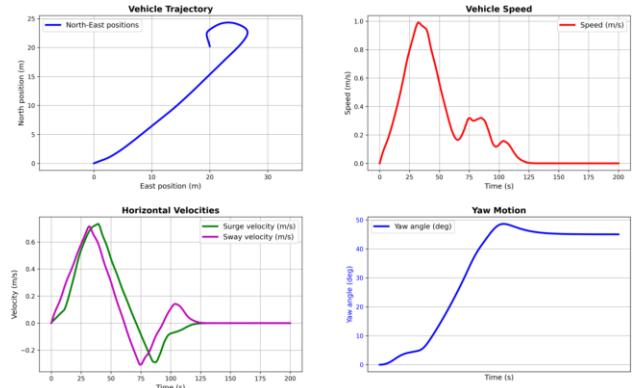
전체 파이프라인은 LangChain과 LangGraph 기반으로 구현되어, 모듈 간 데이터 흐름과 상태가 효율적으로 이루어진다.[3]

III. 연구 결과 및 분석

제안된 RAG 프레임워크의 성능은 데이터베이스에 포함되지 않은 임의의 해상 조건을 설정하고, 추천된 PID 파라미터를 적용한 시뮬레이션을 통해 검증하였다. 해류 방향은 북(0°) 기준 방위각으로 표시하였다.



[그림 2] 해류 속도 1.07 m/s, 해류 방향 133.20°



[Case 2] 해류 속도 1.59 m/s, 해류 방향 50.27°

실험 결과, 해류 속도와 방향이 다른 두 가지 조건에서도 선박은 목표 위치와 선수각에 안정적으로 수렴하였다. 제작은 과도한 오버슈트 없이 안정적으로 수렴했고, 속도와 Yaw 응답 또한 일시적 진동 후 빠르게 안정화되어 강건한 추종 성능을 보였다.

Analysis results of the RAG system -
 ω_{n_surge} : 0.1 rad/s
Reason: A low ω_{n_surge} provides stable longitudinal control and prevents excessive rotation, enabling accurate arrival at the target position. In particular, configurations in the data with high speed (V) and low stability index (Vel) tended to prefer lower rotational rates for more stable control.
 ω_{n_sway} : 0.1 rad/s
Reason: Similarly, a low ω_{n_sway} minimizes lateral oscillations, which is especially important at low speed (V) settings. Reduced lateral oscillation improves energy efficiency and supports more accurate arrival at the target position.
 ω_{n_yaw} : 0.4 rad/s
Reason: This ω_{n_yaw} is moderately high while maintaining stability. In the data, ω_{n_yaw} values between 0.3 and 0.5 received the highest scores. 0.4 rad/s appears to be a balance point that preserves heading-change efficiency while effectively controlling system oscillations.
These settings help optimize system stability and control performance while ensuring accurate arrival at the target position. Notably, at low speeds (V), configurations with a high stability index (Vel) showed superior results.

추가로, 시스템은 단순히 최적 파라미터(ω_{n_surge} , ω_{n_sway} , ω_{n_yaw})를 제시하는 것에 그치지 않고, 값을 선택한 근거와 이유까지 함께 설명하여 결과의 해석 가능성과 실무 적용성을 높였다.

IV. 결론

본 연구는 RAG 기반 AI 프레임워크를 활용해 Frigate Heading Autopilot의 PID 파라미터를 자동 튜닝하는 방식을 제안하였다. 제안된 방법은 대규모 시뮬레이션 로그를 활용한 근거 중심·설명 가능한 추론을 통해 수동 의존을 줄이고, 다양한 해상 조건에서도 강건하고 안정적인 제어 성능을 달성함을 확인하였다.

ACKNOWLEDGMENT

이 논문은 정부(과학기술정보통신부)의 재원으로

정보통신기획평가원-학석사연계 ICT 핵심인재 양성 지원을 받아 수행한 연구임 (IITP-2024-00436744).

참 고 문 헌

- [1] MoMagDii. (2022). uuv_python_simulator: Python Vehicle Simulator [Computer software]. GitHub. https://github.com/MoMagDii/uuv_python_simulator
- [2] Fossen, T. I. (2021). *Handbook of Marine Craft Hydrodynamics and Motion Control* (2nd ed.). Wiley
- [3] Harrison Chase. *LangChain*. arXiv preprint arXiv:2302.07842, 2023.