고려대 소프트웨어 분석 연구 소개

오학주

고려대학교

2025 SIGPL 여름학교

Software Analysis Lab @Korea Univ.

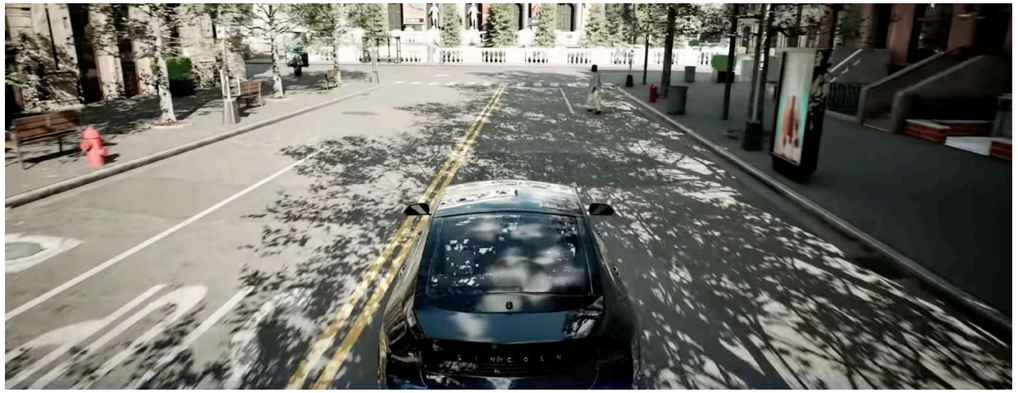
- **Program analysis**
  - Fuzzing, symbolic execution, static analysis, verification
- **Program repair**
  - Static analysis-guided repair, fault localization, LLM-based repair
- **Emerging domains**
  - Smart contracts, Quantum, AI, ADS (Autonomous Driving System)
- **Python infrastructures**
  - Intermediate language, types, compiler correctness, analysis, optimization

자율주행 시스템 (ADS) 테스팅

HW

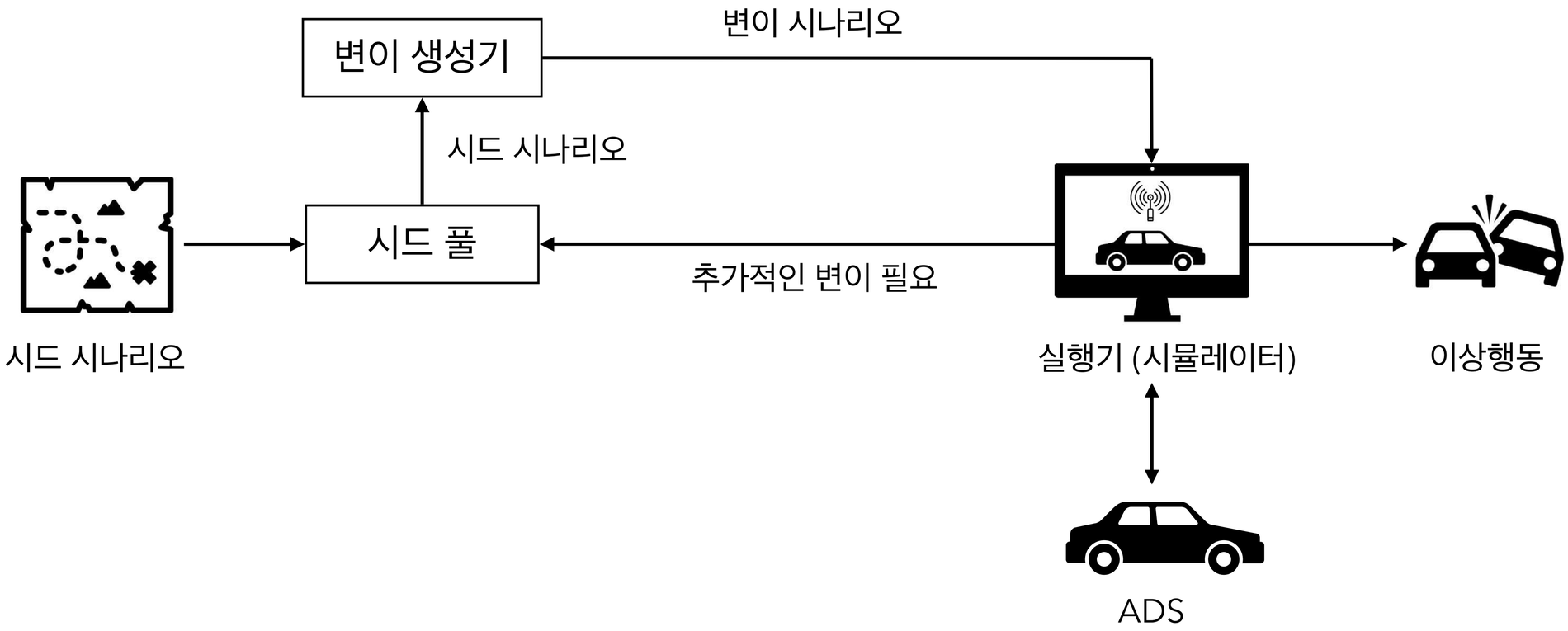
SW

- 대상 ADS: Autoware (https://autoware.org/)
- 시뮬레이터: Carla (https://carla.org/)



자율주행 시스템 (ADS) 테스팅

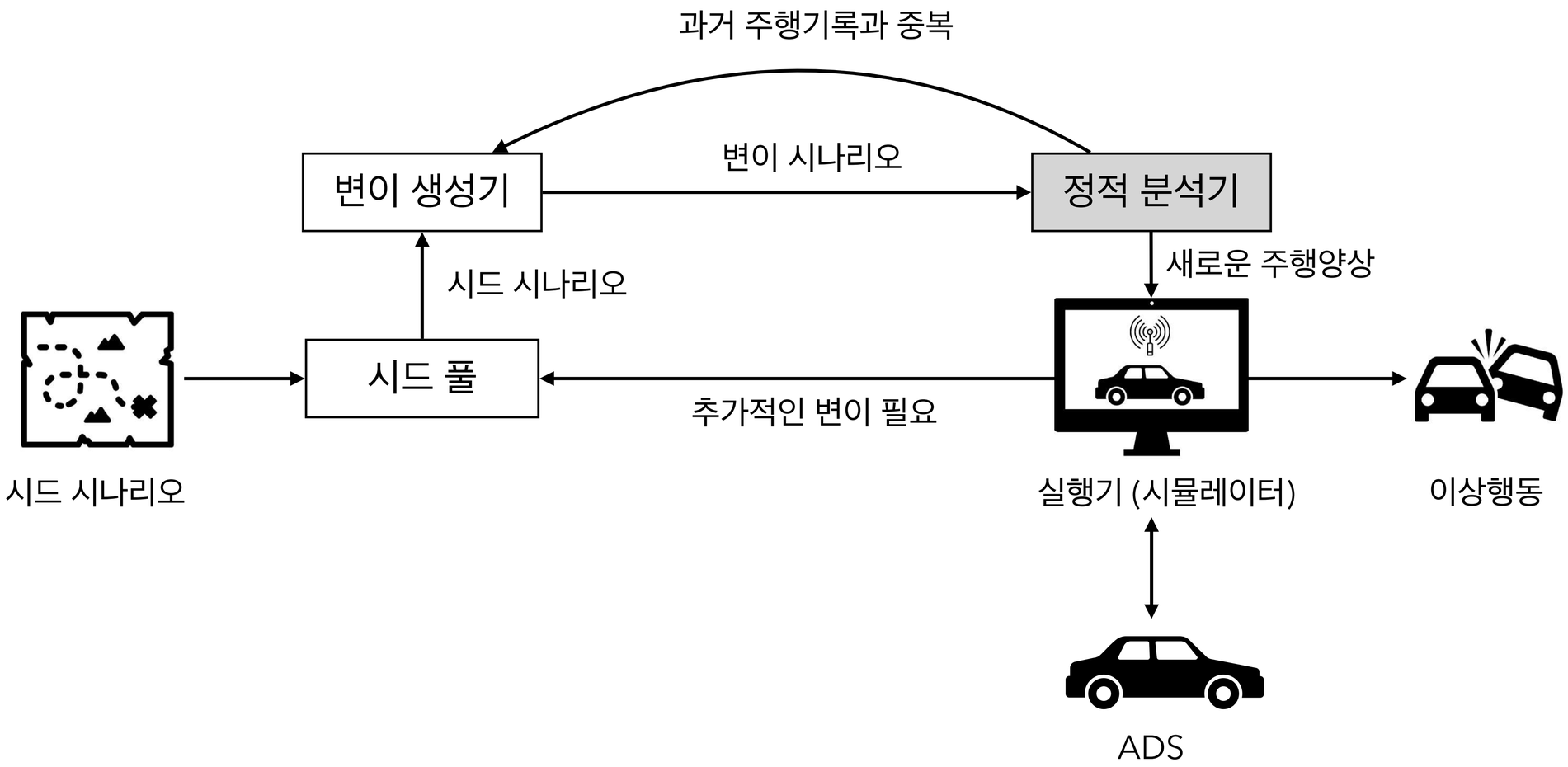
- 통상적인 변이 기반 블랙박스 퍼징



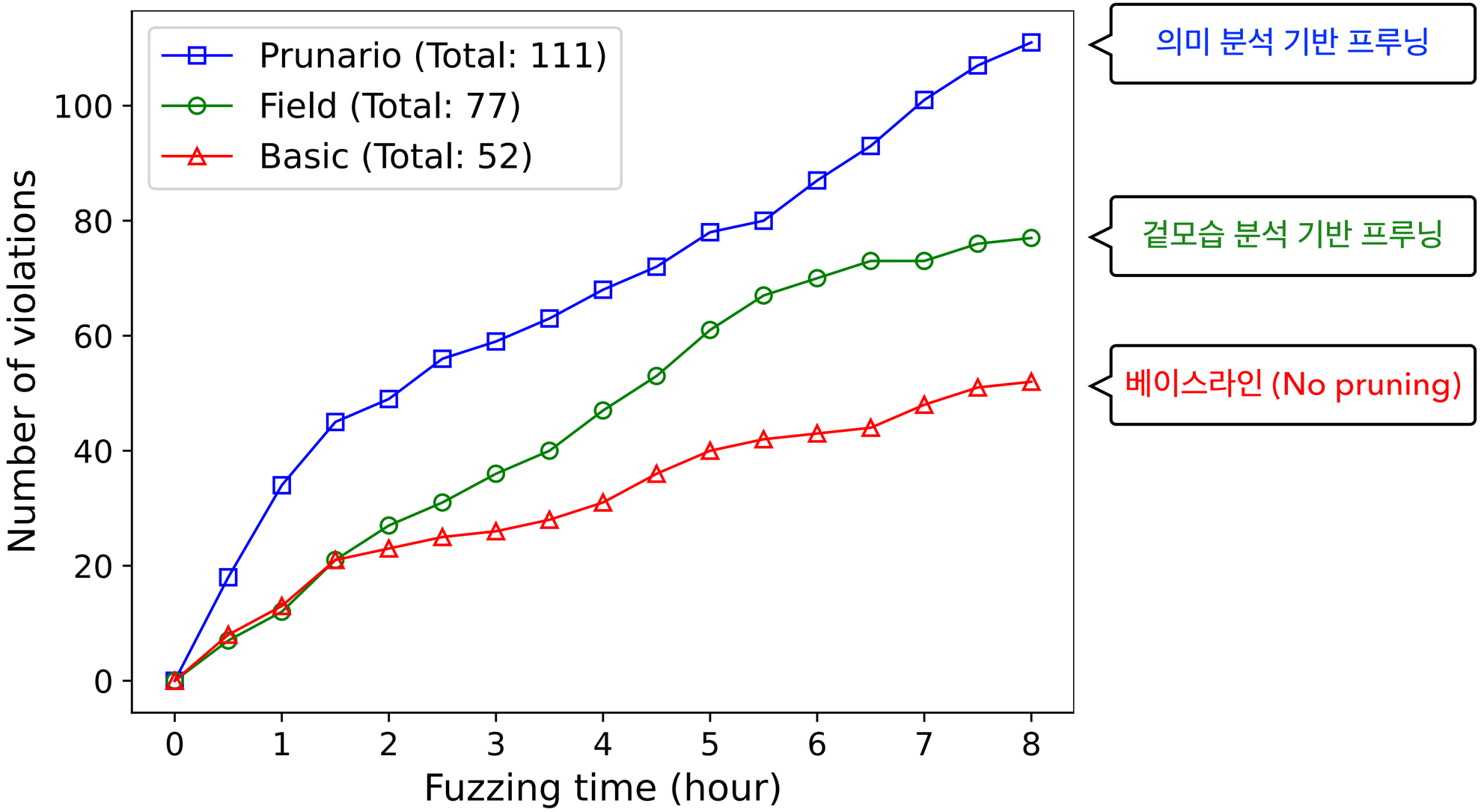
Unique Challenge: High Simulation Cost

아이디어

- “정적 분석”: 시뮬레이터 실행 전에 ADS의 행동을 예측



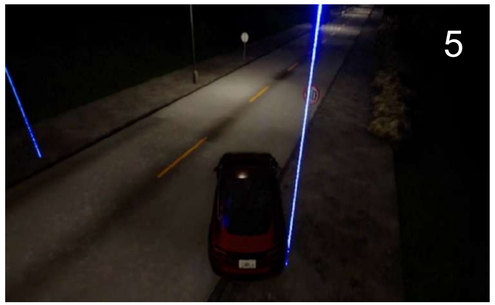
프루닝 성능


Prunario (Total: 111)
Field (Total: 77)
Basic (Total: 52)
Number of violations
0
20
40
60
80
100
0
1
2
3
4
5
6
7
8
Fuzzing time (hour)
의미 분석 기반 프루닝
겉모습 분석 기반 프루닝
베이스라인 (No pruning)

찾은 오류

- 23 reported, 16 confirmed

ID	Commit	Module	Type	Description	ACK
1	f6b14ec	Control	Collision	Weak braking on a downhill.	✓
2	eeed846	Localization	Stalling	Miscalculated the ego vehicle's position while turning.	✓
3	eeed846	Localization	Stalling	Fails to estimate the ego vehicle's position on a slippery surface.	✓
4	eeed846	Planning	Stalling	Indefinitely stops while making a turn at a sharp corner.	



13	4a3de49	Localization	Stalling	Halts unnecessarily due to localization error at roundabout.	✓
14	4a3de49	Perception	Stalling	Detects a non-blocking object as an obstacle ahead.	✓
15	4a3de49	pending	Collision	Fails to consistently detect a small obstacle in front.	✓
16	4a3de49	Planning	Collision	Incorrectly calculated a drivable area ahead as non-drivable.	
17	4a3de49	Planning	Stalling	Incorrectly recognize the slowly approaching vehicle as moving away.	
18	4a3de49	pending	Collision	Fails to yield to the NPC vehicle exiting from a roundabout.	✓
19	4a3de49	Planning	Stalling	Underestimates required acceleration for uphill.	✓
20	4a3de49	Control	Stalling	Stops near the destination at the corner.	✓
21	4a3de49	pending	Stalling	Remains stopped despite the obstacle ahead having cleared.	✓
22	4a3de49	pending	Stalling	Failed to drive downhill when initialized close to the slope start.	
23	75549a6	pending	Invasion	Produces an invalid backward plan instead of a valid forward route.	✓

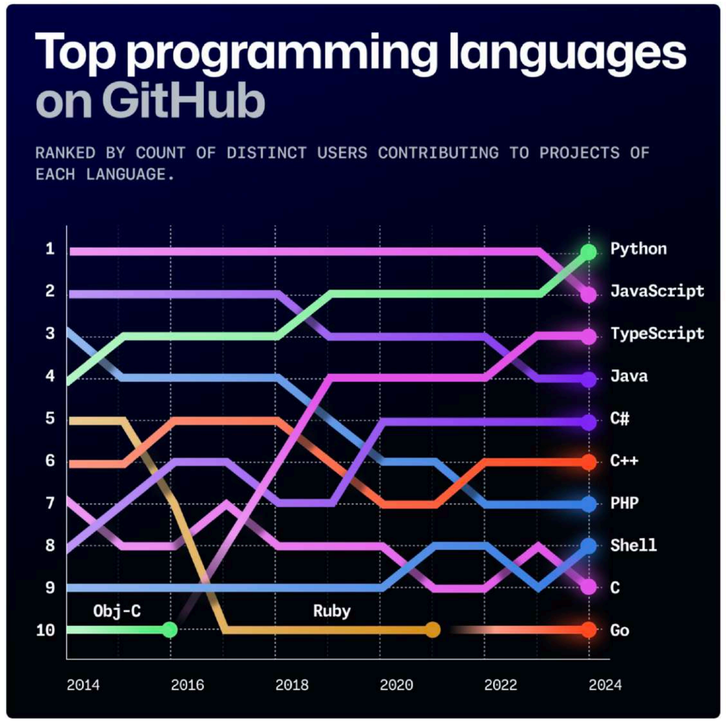
확장 계획

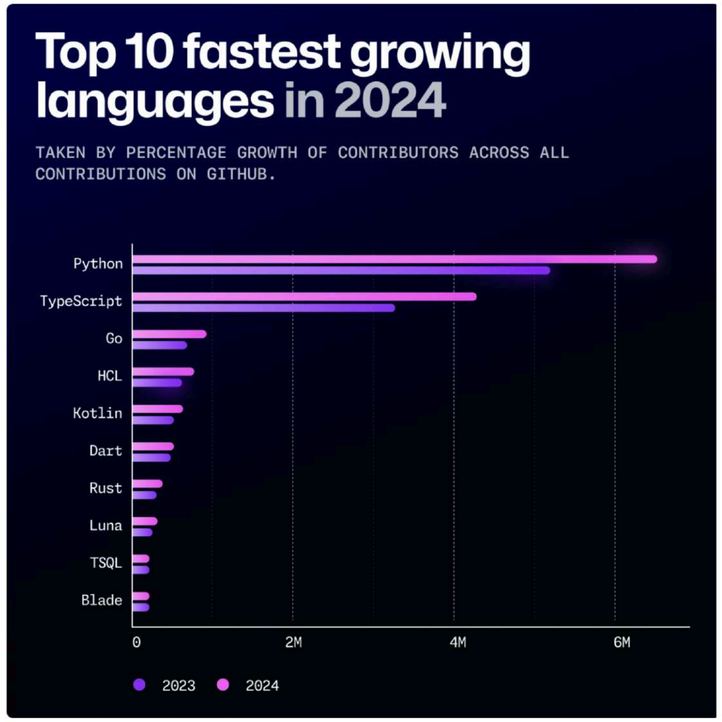
- 정적 분석 정확도 향상
- 다른 Physical AI 시스템 테스팅으로 확장: 로봇, 드론, etc

Python

- 산업 전반에 걸쳐 핵심 프로그래밍 언어로 도약 중
  - 웹, 과학 계산, 금융, AI 에이전트, 로봇, 고성능 시스템, 고신뢰 시스템, …

⇒ Python 코드의 안전성·성능·이식성 확보 기술 필요





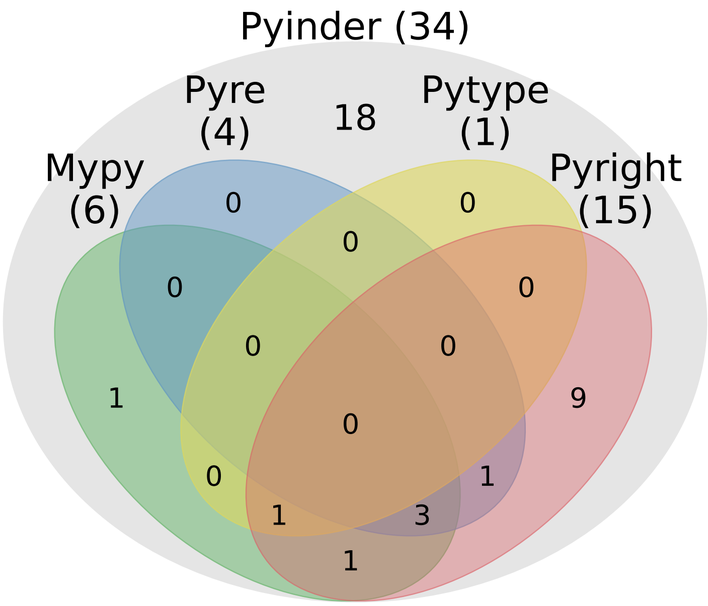
Our Research on Python

- PyTER: Effective program repair for Python type errors. FSE 2022 — 안전성
- Towards effective static type-error detection for Python. ASE 2024 — 안전성
- Boosting Python type inference models. In submission — 안전성
- Automating Python library migrations. In submission — 이식성
- Python intermediate language. In progress — 안전성, 이식성
- Python compiler testing. In progress — 안전성

Static Types for Python

- 타입 오류 검출
  - Mypy, Pyre (Meta), Pytype (Google), Pyright (MS)
  - 제한적 성능: Pyright 오류 탐지율은 20% 수준
- 정적 타입 추론

```
def add(x: <FILL_IN>, y: <FILL_IN>) -> <FILL_IN>:
    return x + y
```



Fundamental Question: 동적 언어의 타입을 정적으로 예측한다는 것이 무엇인가?

오원석 (15분)

Python 중간 언어

이석현 (15분)

- Python 코드의 안전성·성능·이식성 확보 필요. 하지만 정확한 분석, 최적화, 변환이 매우 어려운 상황

As-Is

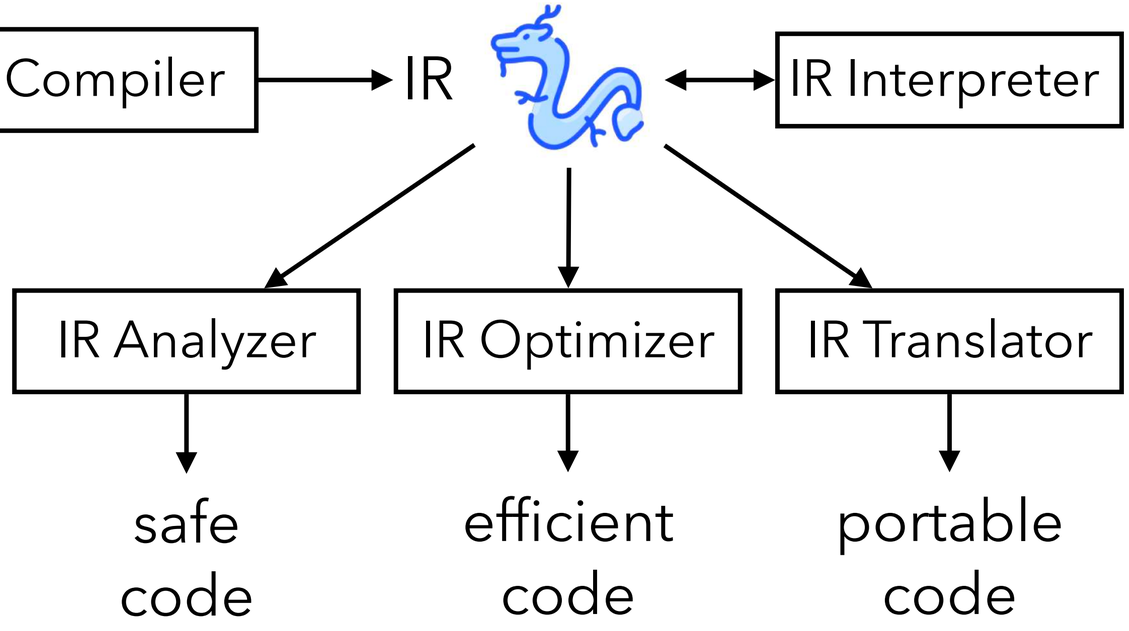
Informal, complex, and implicit

Python → Python-to-IR Compiler → IR

- Hard to analyze → unsafe code
- Hard to optimize → inefficient code
- Hard to translate → unportable code

To-Be

Formal, minimal, and explicit



정리

- Physical AI 테스팅
- Python 타입 분석 및 추론
- Python 중간 언어 인프라

피드백 / 공동 연구 환영합니다!