

PLDI 2025

» SEOUL, SOUTH KOREA



HYUNJOON KIM

JUNE 2025

INTRODUCTION

PLDI는 프로그래밍 언어 분야에서 POPL과 OOPSLA와 더불어 최고 수준의 학회라고 할 수 있다. 공교롭게도 올해 PLDI는 우리나라 서울에서 열려 논문 제출을 하지 않았음에도 연구실의 지원을 받아 참석할 수 있었다. 참석할 수 있는 기회를 주신 박지혁 교수님께 감사드리며, 이 글을 통해 학회에서 경험하고 느낀 내용들이 잘 전달되고 도움이 되길 바란다.

이번 PLDI'25의 main session은 18일부터 20일까지 총 3일에 걸쳐 진행되었고, 그 전 16일과 17일에는 함께 열린 workshop과 tutorial들이 진행되었다. 나는 월요일부터 금요일까지 5일 내내 참석할 수 있었고, 월요일과 화요일에는 egglog를 이용해 간단한 분석기와 최적화를 구현하는 튜토리얼, 그리고 SOAP와 RPLS workshop의 talk들을 주로 들었다.

태어나서 처음으로 참가하는 국제 학회였기에 기대가 크면서도, 한편으로 talk들을 잘 이해할 수 있을까 하는 걱정도 컸다. 솔직히 관심가는 발표들을 만족할 만큼 전부 이해하는 것은 불가능했고, 돌아와서 논문을 뒤적이며 내용을 보충해야 했다. 하지만 다양한 학생들과 만나 이야기하며 동기부여를 받기도 하고, 연구하며 힘들었던 지점들을 나눴던 경험 자체가 꽤나 즐거웠다. 낯선 외국에서 개최되었다면 시차 적응으로 talk들에 집중하기 어려웠을텐데, 서울 그것도 학교에서 가까운 곳에 학회장이 위치해 정말 운이 좋았다는 생각이 들었다.

학회 기간 동안 talk들을 들으면서 계속 듣 생각은 내가 알던 것보다 PL에는 훨씬 더 많은 연구 분야와 주제들이 있다는 것이었다. 학교에서 뵙던 교수님들이 주로 정적 분석을 연구하시다 보니 나도 자연스럽게 분석 연구들만 생각하고 있었는데, 분석 외에도 검증이나 타입 시스템과 같은 주제들, 그리고 확률론적 프로그래밍과 양자 컴퓨팅과 같은 생소한 도메인들을 대상으로 한 다양한 연구들을 볼 수 있었다. 최근에 새롭게 시작한 주제가 퍼징이었기에, 테스팅이나 퍼징에 대한 발표들이 있다면 참석하려 했지만 아쉽게 많지 않았다. 다만 그 외에도 흥미를 끄는 talk들은 충분히 많아, 이 글에서는 그러한 발표들 중에서 특히 인상적이었던 몇 개를 소개하고자 한다.

INTERESTING TALKS

RELAXING ALIAS ANALYSIS: EXPLORING THE UNEXPLORED SPACE

프로그램을 최적화할 때 우리는 의미를 깨뜨리지 않고 더 나은 최적화를 하기 위해 분석을 진행한다. 전통적으로 “더 정확한 alias 정보가 더 나은 최적화로 이어진다”는 가정이 지배적이고, 사실은 직관적으로 그럴 법한 가정인데, 이 발표는 그러한 전제 자체에 도전하여 매우 신선했다.

이 연구에서는 정확한 alias 정보를 의도적으로 relaxing하는 접근을 통해 LLVM에서 alias analysis의 실제 영향력을 측정했다. 놀랍게도 그 결과는 기존의 예상을 뒤집는 내용이었다. 모든 alias analysis를 제거하더라도 최종 바이너리에 미치는 영향은 미미했고, 컴파일러가 요청하는 alias query 중 단 3% 만이 실제 최적화에 기여하고 있었다. 더욱 흥미로운 점은 일부 경우에 무작위 relaxation이 오히려 실행 시간을 21% 까지 줄이고 더 작은 바이너리를 생성했다는 것이다.

이러한 결과를 바탕으로 컴파일러 연구의 방향성이 단순히 분석을 통해 더 많은 정보를 수집하는 것보다 실제로 영향력 있는 정보를 식별하고 효과적으로 활용하는 방향으로 나아가야 한다는 이야기로 귀결되었는데, 좋은 연구가 꼭 새로운 방법론을 제안해야만 하는 것은 아니라는 점을 다시 한 번 깨닫게 해 준 발표였다.

이 발표를 듣고 교수님과 이야기하면서, 교수님께서는 우리 연구에도 비슷한 질문을 던져볼 수 있겠다는 이야기를 하셨다. 우리가 당연하게 받아들이고 넘어가던 전제들을 다시 확인해보면 어떨까 하는 것이었다. 발표를 들으면서 정작 내 연구의 관점에서 얻을 수 있는 교훈들을 생각하지 못하고 내용을 따라가기 급급했었는데, 이런 연결점을 찾아내는 것이 학회 발표를 제대로 듣는 방법이라는 것을 배우는 순간이었다.

TREE BORROWS

RPLS에서의 Minirust 발표를 비롯해 이번 학회에서는 러스트 관련 발표가 상당히 많았다. 역시 작년 Programming Languages Software Award의 주인공답다는 생각이 들었다.

특히 인상적이었던 것은 Ralf Jung 교수님의 Tree Borrows 발표였다. 지인들이 학회 전부터 꼭 들어보라고 추천한 발표였는데, 그의 POPL'20 논문인 Stacked borrows의 후속작이라고 했다. 이전 논문을 읽어본 적이 없어 걱정했지만, 다행히 기존 연구도 함께 설명해주어 전체적인 흐름을 이해할 수 있었다.

이 연구가 다루는 문제는 매우 실용적이면서도 이론적으로 흥미로웠다. 러스트에서 mutable reference는 서로 다른 메모리를 가리켜야 한다는 규칙이 있고, 컴파일러는 이를 믿고 최적화를 수행한다. 예를 들어:

```
fn write_both(x: &mut i32, y: &mut i32) → i32 {  
    *x = 13;  
    *y = 20;  
    *x // 컴파일러는 *x를 메모리에서 다시 읽지 않고 13을 직접 반환  
}
```

그런데 unsafe 코드는 이러한 가정을 깨뜨릴 수 있어, 어떤 aliasing이 undefined behavior인지 명확히 정의해야 한다. 이전의 Stacked Borrows는 reference 관계를 stack으로 관리했지만, 이번 Tree Borrows는 tree 구조를 사용해 더 많은 안전한 aliasing 패턴을 허용하면서도 컴파일러 최적화의 정확성을 보장한다.

이 발표를 들으며 특히 인상적이었던 점은 실용적 필요와 이론적 엄밀함을 동시에 추구하는 연구 방향이었다. 단순히 제약을 강화하는 것이 아니라, 실제 프로그래머들이 작성하는 패턴을 더 많이 수용하면서도 안전성을 보장하는 방법을 찾는 것. 이러한 접근이 프로그래밍 언어 연구가 실무에 기여하는 좋은 예시라는 생각이 들었다.

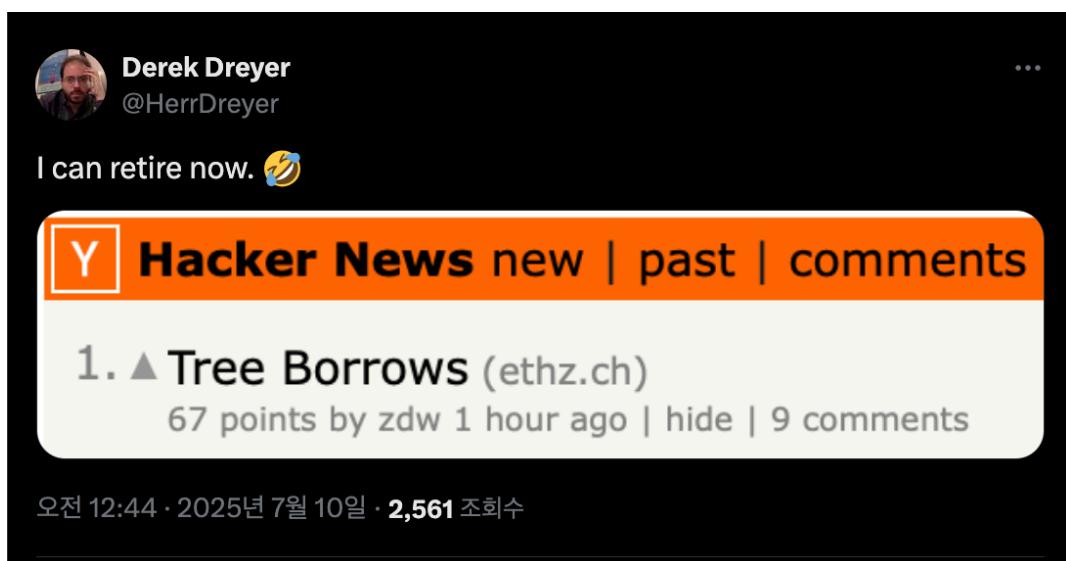


Figure 1: 여담으로, 이 글을 정리하는 7월 10일 기준으로 Tree Borrows가 Hacker News 1위에 올랐다.

OPTIMIZATION-DIRECTED COMPILER FUZZING FOR CONTINUOUS TRANSLATION VALIDATION

이 발표는 개인적으로 특별한 의미가 있었다. 학부 때부터 친하게 지낸 재성이 형이 석사과정에서 진행한 연구였기 때문이다. SIGPL 때나 대전에 놀러갈 때마다 연구 과정에서 마주한 어려움들을 들어왔는데, 그 노력들이 PLDI 논문으로 결실을 맺고 공동 연구자인 장봉준님의 발표로 듣게 되니 감회가 새로웠다.

이 연구가 다루는 문제는 명확하면서도 실용적이다. 번역 검산기 (Translation Validator)는 컴파일러가 최적화 후에도 프로그램의 의미를 보존하는지 검증하는 중요한 도구다. 하지만 검산기는 주어진 입력에 대해서만 검증할 수 있는데, 복잡한 최적화를 실제로 trigger하는 입력을 만들기가 매우 어렵다는 근본적 한계가 있다.

이 연구는 그러한 한계점을 지향성 퍼징으로 돌파한다. 먼저 컴파일러 소스코드를 분석해 최적화 관련 경로를 추출하고, 해당 경로를 목표로 퍼징을 수행하여 특정 최적화를 trigger하는 프로그램을 자동으로 생성하는 방식이다. 실제로 이 접근법으로 최신 LLVM에서 55개의 새로운 miscompilation 버그를 발견했다.

발표를 들으며 다시금 깨달은 것은 “좋은 연구는 또 다른 좋은 연구의 토대가 된다”는 점이었다. 번역 검산이라는 의미 있는 기술이 먼저 있었고, 그 한계점이 명확했기에 지향성 퍼징이라는 새로운 접근이 가능했다. ESMeta 위에서 다양한 연구들이 계속 이어지는 우리 연구실처럼, 좋은 토대가 있으면 그 위에서 더 많은 것들이 가능해진다는 걸 다시 한 번 느꼈다.

PRACTICAL TYPE INFERENCE WITH LEVELS

타입 추론에 대한 이 발표는 PL 연구의 고질적인 문제를 정면으로 다뤘다. 이론적으로 아름다운 specification이 실제 구현에서는 비효율적인 경우가 많다는 것이다. 대표적인 예가 let-generalization인데, 교과서대로라면 전체 typing context를 순회해야 하지만 이러한 구현은 성능 문제가 있다.

이 연구가 제시한 해결책은 단순하고 깔끔했다. 모든 type variable에 ‘level’이라는 값을 부여하고, “Level n 이하의 context에 있는 type들이 level n+1의 variable을 참조할 수 없다”는 단순한 규칙으로 문제를 해결하였다. 타입 체킹에 일종의 scope를 도입한 셈인데, 이 간단한 아이디어로 let-generalization부터 higher-rank polymorphism까지 다양한 기능을 효율적으로 구현할 수 있게 되었다.

발표 중 polymorphic promotion 부분은 온전히 이해하지 못했지만, 핵심 아이디어에 집중할 수 있어서 좋았고 필요할 때 논문을 다시 찾아볼 좋은 이유가 생겼다. 좋은 추상화는 복잡해 보이는 여러 문제를 단순하고 통합된 방식으로 해결할 수 있다는 것을 느끼게 한 발표였다.

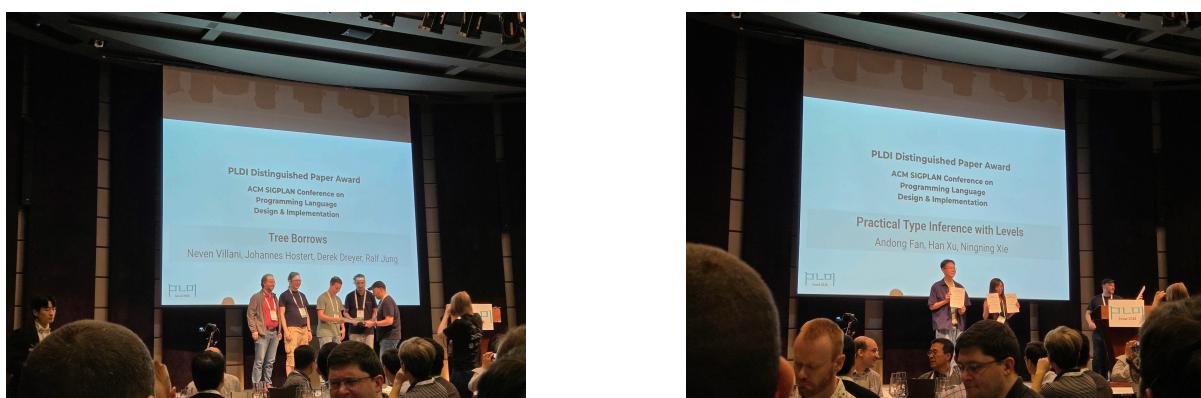


Figure 2: PLDI'25의 distinguished papers들 중 일부

BEYOND THE TALKS

쉬는 시간에는 커피와 다과가 제공되었고, 점심이나 저녁 식사도 굉장히 맛있었다.



Figure 3: 맛있는 밥은 언제나 중요하다

뱅큇 전후에는 지향성 퍼징 분야에서 연구를 하고 계시는 우리 학교 오학주 교수님 연구실의 변지석님과 KAIST 허기홍 교수님 연구실의 박건님과 같이 이야기를 나눴다. ESMeta를 만지면서 틈틈히 퍼저를 손보고 고친 경험은 있지만, 본격적인 퍼징 연구 경험과 배경지식이 부족했던 터라 두 분과의 discussion이 특히 유익했다. 대화를 나누며 자연스럽게 크고 작은 팁들을 얻을 수 있었다. 또한, 기존에 알고 지내던 지인들과 오랜만에 만나서 요즘은 어떤 연구를 하고 있는지 이야기를 나눌 수 있어 좋았다. 특히 이들이 자신의 또 다른 지인들을 소개해주면서 네트워킹의 폭이 자연스럽게 넓어졌다.

이렇게 만난 해외 연구자들과 대화하다 보니, 아직 초기 단계지만 내가 진행 중인 연구와 관심 topic에 대해 영어로 설명하는 좋은 연습이 되었다. 다음 학회에서는 더 많은 외국 연구자들과 적극적으로 대화하고 싶다는 욕심이 생겼다. 네트워킹의 기회를 만들어준 서울대 허충길 교수님 연구실의 순원이 형에게 감사드린다.

CLOSING THOUGHTS

PLDI'25를 통해 좋은 연구와 좋은 발표가 무엇인지 배웠다. 당연시하던 전제를 의심해보는 것, 이론과 실무를 함께 보는 것, 그리고 이를 명확하게 전달하는 것의 중요성을 깨달았다. 특히 다양한 발표 스타일을 관찰하며 ‘이해하기 쉽고 전달력 있는 발표’의 기준을 세울 수 있었던 것이 큰 자산이 되었다. 이러한 경험은 향후 내 논문을 쓰고 발표하는 데에 중요한 reference가 될 것 같다.

지원해주신 박지혁 교수님께 다시 한번 감사의 말씀을 드리며 이 글을 마친다.