

# Trip Report

POPL/VMCAI 2011  
Austin, Texas, USA

최원태 [wtchoi@ropas.snu.ac.kr](mailto:wtchoi@ropas.snu.ac.kr)  
서울대학교 프로그래밍 연구실

텍사스 오스틴에서 열린 POPL/VMCAI 2011 학회에 참석하였다. 지도교수님이신 이광근 교수님과 같은 연구실에서 박사과정을 하고있는 학주형, 석사과정인 성근이형, 우석이와 함께 한 여정이다. 우리 연구실에서는 학주형이 VMCAI에서 논문을 한편 발표하였고, 나 또한 POPL에서 논문을 한편 발표하였다. 공교롭게도 두 사람다 스페인 마드리드에서 열린 작년 POPL/VMCAI때는 참관자 신분으로 참여했다. 달라진 입장에서 경험한 학회는 새로웠다. 우선 가는 비행기 안에서 부터 리허설을 했으니까. 작년에는 마드리드 가이드북을 보던 우리였다.

학회장소인 Omni Hotel은 유리로된 네모꼴의 건물 안에 삼각형모양의 작은 건물 두개가 들어가있는 형상이었다. 두 작은 건물 사이는 유리천장으로 가려졌지만 태양광으로 채광되고 위가 뚫린 실내광장이 자리하고 있었다. 놀라운 것은 그 넓은 실내광장을 에어컨디셔닝 하고 있다는 점이다. 미국답다는 말 이외에는 적합한 표현을 찾지 못했다.

## 1. POPL'11 논문 발표

---

### 논문

POPL'11 첫날 점심에 "Static Analysis for Multi-Staged Programs via Unstaging Translation" 논문을 발표하였다. 논문에서는 다단계 프로그램(프로그램 실행중에 프로그램 코드가 생성되고 생성된 코드가 다시 실행되는)을 의미가 동일한 단계없는 프로그램으로 변환하여 분석하는 내용을 다루고 있다. 이 논문은 터키 Ozyegin 대학의 Baris Aktemur 교수, 지도교수님이신 이광근 교수님, 일본 NII의 Makoto Tatsuta 교수님과 함께 작성하였다. 논문은 약 2년전에 UIUC의 박사과정이던 Baris가 자신의 박사논문 원고를 이광근 교수님께 보내면서 시작되었다. 당시 나는 다단계 언어를 위한 정적분석을 고안하다가 막다른 골목에 다다른 상태였다. Baris의 논문을 보고 눈이 번쩍했다. Baris는 "타입을 보존하는 변환"을 이미 고안한 상태였는데 조금만 고치면 실행의미도 보존하게 만들수 있을것 같았다. 이광근 교수님께서 Baris를 초청하면서 협력이 시작되었다. 1년 반에 걸친 고투 끝에 논문을 완성할수 있었다. Baris의 어깨에 올라타고 이광근 교수님의 지휘를 따라가서 고지에 올랐으니 나는 운이 좋은 편이다.

### 발표준비와 발표

처음 해보는 학회 발표였다. 너무 긴장해서였는지 발표순간은 잘 기억나지 않는다. 머리를 비우고 리허설했던 내용을 리플레이하고 내려왔다. 한가지 기억나는 것은 세번째 슬라이

드를 발표할 즈음 연구실 동료들과 Makoto 교수님이 갑자기 웃음을 터뜨린 것. 발표중에는 뭔가 실수를 했나 싶었다. 후에 이우석 학생의 증언을 들어보았다. “목소리와 행동이 너무도 차분하여 처음에는 전혀 긴장하지 않은 줄 알았어요. 스크린상에 비친 레이저 포인터 사시나무 떨리듯 흔들리는 것을 보고 역시나 했어요” 큰 실수는 없었나보다.

긴장했음에도 발표를 무사히 마칠 수 있었던 것은 발표 자료에 대한 자신감 덕분이었다. 내가 영어를 못해도 이 발표자료라면, 그리고 POPL 에 참여하는 사람들이라면 알아들을 것이라고. 이광근 교수님과 발표자료를 보름정도 두들겼고, Baris에게도 두차례에 걸쳐 긴 피드백을 받았다. 간단히 넘어가도 되는 부분과 자세한 설명이 필요한 부분은 어디인지를 알았고 그에 따라 발표도 준비할 수 있었다. 따로 대본을 만들지 않았다. 대신 직접 말하면서 녹음해보고 빠르게 들어보면서 표현이 이상한 부분을 찾고 시간을 조절했다. 한번에 한시간 정도 걸리는 작업이었는데, 세번쯤 반복하면 한결 나아졌다는 느낌을 받을 수 있었다. 학회에서 편안하게 발표하듯 보였던 선배들은 발표전에 얼마나 노력했을까를 생각하니 세삼 대단하게 느껴졌다.

발표가 끝나고 다섯명 정도가 질문을 했다. 이때부터 정신이 돌아왔는데, 다단계 언어에 관심을 가진 사람이 생각보다 많아 한번 놀랐고, 질문을 듣기가 쉽지 않아 다시 한번 놀랐다. 끝내 질문을 알아들을 수 없었던 두명에게는 답변이 길어질 듯 하니 세션이 끝나고 만나자고 얼버무렸다. 다행히도 세션이 끝난 후에는 제대로 답을 해줄 수 있었다. 그중 한명은 Scheme에 관련된 연구를 하는 학생인데 처음에 같은 어휘를 서로 다른 어휘로 사용하여 의사소통에 어려움이 있었다. 서로 vocabulary를 맞춰가면서 의사소통을 이루어나가는 것도 재미있는 경험이었다.

## 느낀 것

- 우선 내용을 확실하게 소화해야 한다. 당연하게 들릴 수도 있다. 그렇지만, 연구를 하는 도중에는 의외로 세심한 디테일을 점검하지 않고 넘어가는 일이 많다. 논문을 쓸 때 까지도 교수님과 동료들의 도움을 받거나 이해하지 못한 부분을 얼버무리는 식으로 넘어갈 수가 있다. 하지만 발표 때는 도망칠 수 없다. 자료를 만들 때, 처음에는 도무지 25분짜리 슬라이드로 내용이 압축되지 않아 당혹스러웠다. 준비를 하면서 내가 낸 아이디어라고 내가 소화하고 있는 것은 아닐 수도 있으며, 소화하지 못하면 25분 안에 전달할 수 없다는 사실을 알았다. 어떻게 설명할지 고민하고 제대로 이해하지 못한 디테일을 채우기를 반복하다가 출발 4일 전쯤 비로소 필요한 디테일은 다 들어있으면서 상황하지 않다는 느낌이 들었다. 차이는 자신의 아이디어를 둘러싼 디테일들을 온전히 이해하고 있느냐에 있는 듯 했다.
- 또 한가지. 듣기가 잘 되어야 한다. 긴장감 때문인지 아니면 긴장풀린 때문인지, 발표 직후 사람들의 질문을 잘 알아들을 수가 없었다. 연구자들은 대체로 심성이 착해서 느리게 말하고 쉬운 단어만 써서 더듬더듬 말해도 뜻이 통하면 잘 참아준다. 그런 반면에 본인이 하는 말이 알아듣기 쉬운지 어떤지는 별로 신경쓰지 않는다. 제대로 듣고 그에 대해 제대로 된 내용으로 답을 한다면 틀린 영어를 사용하더라도 일단 대화는 이루어진다. 듣는 능력이 중요하다. 다양한 억양으로 다양한 빠르기로 하는 말을 일단 알아들을 수 있어야 커뮤니케이션이 올바르게 이루어질 것 같다.

## 2. 보고 들은 것들

---

### 한국의 약진

이번 POPL/VMCAI에서는 한국인의 발표가 각각 두팀이 있었다. VMCAI에서는 우리 연구실의 학주형과 카이스트의 김세원 박사님이 발표를 하셨다. 학주형은 Local Reasoning을 적용해서 실행흐름을 고려하는 함수간 분석의 비용을 획기적으로 줄이는 내용을 발표하였고 김세원박사님은 String Analysis를 오토마타연산을 사용해 요약해석틀 안으로 끌어들이는 이론적인 성과를 발표하였다. POPL에서는 MPI에 계시는 허춘길박사님이 ML프로그램과 그 컴파일 결과인 바이너리 프로그램 사이에 contextual equivalence가 있음을 보인 내용을 발표하셨다. 내 발표 바로 다음인데 방이 달라서 우리 연구실 사람들은 우루루 이동해야했다. 직접 발표하지는 않았지만 양홍석 박사님도 POPL에 논문을 한편 내셨다.

### VMCAI

- **Access Nets : Modeling Access to Physical Spaces** : 건물의 접근관한및 관리 정책이 의도한대로 작동하는지를 정적분석 기법을 사용하여 검증하는 내용을 소개했다. 어느 방에 몇명의 사람이 들어있는지가 프로그램의 상태에 대응되고, 방의 연결상태와 관리정책이 상태전이 함수에 대응된다. 문제의 크기를 줄이기 위해 주어진 건물을 “의미가 동일” 한 다른 건물로 변환하기도 하고, constant propagation과 비슷한 아이디어를 사용하기도 하는 등 여러가지로 흥미로웠다. 관리정책과 건물의 생김을 xml로 옮기는 번잡스러운 과정에 대한 해법을 제시하지 않았다는 점에서 완성도가 높지는 않았다.
- **Proving Stabilization of biological systems** : 이 논문에서는 세포가 가질수 있는 상태를 정적분석으로 추적하는 내용을 다루고 있다. 핵심 아이디어는 Local Reasoning을 이용해서 세포 단위로 상태를 분석하고 그렇게 모인 정보를 인접한 세포들에게 전달하여 다시 분석하는 것이다. 생명체를 구성하는 세포들은 외부환경과 가능한 독립적으로 작동하도록 진화해 왔기 때문에 Local Reasoning이 아주 잘 어울린다고 한다. 생명체를 분석하는데 Semantics를 정의하고 정적분석을 돌린다니 기분이 이상했다. 이광근교수님께서 종종 “Syntax와 Semantics를 정의할 수 있으면 프로그램을 짤 수 있는 것이고 분석도 가능하다”는 이야기를 하시는데 그것이 얼마나 강력한 아이디어인지 세삼 느껴지는 발표였다.
- **Distributed and Predictable Software Model Checking** : 이 논문은 CEGAR기반 모델체킹을 병렬계산을 이용해 빠르게 만드는 기법을 소개한다. 모델체킹의 경우 하나의 모델에서 다양한 반례를 찾을 수 있다. 또한 어떤 반례를 사용해 모델을 정제하느냐에 따라 분석기 빠른 시간에 끝나기도 하고 메모리부족으로 죽을수도 있다. 그리고 통상적으로 반례가 간단하면 분석이 끝날 확률이 높다고 알려져있다. 이 논문에서는 여러 노드에서 각자 다른 반례를 찾도록 한 다음 그중 가장 간단한 반례를 이용해 모델을 정제한다. 노드의 수가 많을 수록 더 간단한 반례를 찾을 가능성이 크기 때문에 노드가 증가

할수록 검증시간은 점점 빨라지게 된다. 쉬는시간에 발표자에게 이것저것 물어보면서 제대로 이해할수 있었다. 비싸서 할 수 없었던 계산을 병렬적으로 빠르게 수행함으로써 전체계산을 더 빠르게 만드는 아이디어가 내게는 새로웠다. 정적분석에도 그런 여지가 있지는 않을까.

## POPL

작년과 다르게 이번 POPL에서는 프로그램 분석 관련 논문의 약진이 두드러졌다. 특히 첫재날에는 Patrick Cousot 교수가 발표마다 질문을 하는 모습이 정적분석학회인 SAS를 연상시켰다. Verified Compiler 관련된 논문이 많았던 것도 인상적이었다. 무엇보다도 학회를 시작하는 첫번째 Invited talk이 Xavier Leroy의 Verified Compiler 연구개괄이니까. Type을 위한 학회라고 불리던 과거와는 다르게 Verification이 점점 치고올라오는듯해서 기분이 좋았다.

들었던 많은 발표중에 "민감성" 이라는 테마로 묶을 수 있는 논문 네편을 소개한다. 분석 기술이 점점 발전하면서 함수간의 호출관계와 실행흐름 고려하기 시작했다. 호출 관계를 고려하는 분석은 "호출문맥민감" 분석이라 부르고, 실행흐름을 고려하는 분석을 "실행흐름에 민감한" 분석이라고 부른다. 호출문맥이나 실행흐름을 정직하게 따라가면 너무 분석 비용이 비싸다. 그렇다고 무시하면 정확도가 떨어진다. 적당한 중간지점을 찾아야한다. 이번 POPL에 나온 분석과 관련된 논문들중에 중간지점을 찾고자 하는 논문들이 다수 있다. 그리고 그 결과물은 그 동안 많은 사람이 "최대한의 민감함"을 쓴 값에 지원하는 방법을 찾는데 집중해 온 노력을 무색하게 만든다. 발표를 들으며 프로그램분석 분야가 점점 발전하고 있다는 확신이 들었고 아직 할일이 많다는 생각도 들었다. 무술의 달인들은 평상시에는 힘을 빼고 있다가 정확하게 필요한 만큼만 힘을 사용한다고 하는데 분석분야는 이제 막 그러한 필요성을 배운듯 하다. 개인적으로 석사를 시작하면서부터 관심이 있었던 주제라 아주 흥미로웠다.

- **Pick Your Contexts Well : Understanding Object-Sensitivity** : 절차형 언어로 작성된 프로그램을 분석할 때는 호출지점을 이용해서 호출문맥을 구분한다. 객체지향 언어에는 또 다른 옵션이 있다. 객체지향 언어에서 함수는 객체에 종속되어있으므로 호출된 함수를 소유한 객체를 통해 호출문맥을 구분하는 것이다. 이런 아이디어를 **object-sensitivity** 라고 부른다. 이 논문에서는 객체를 보는 대신 객체의 타입 보는 아이디어를 제시한다. 타입을 사용하면 경우의 수가 줄어들어서 분석이 빨리 끝나게 된다. 흥미로운 점은 정확도는 큰 차이가 없더라는 것이다. Java프로그램을 계속해서 분석하던 사람이 낼 수 있는 아이디어인듯했다. 발표를 듣고 숙소에 돌아와 생각하면서 아이디어의 단순함과 효과에 감탄하였다. 관찰이 면밀하고 정확할수록 아이디어는 간단하면서 강력해지는 것 같다.
- **Learning Minimal Abstractions** : 모든 함수가 동일한 만큼 문맥을 구분하면서 분석될 필요가 있는 것은 아니라는 것은 분석을 해본사람이면 누구나 알고있을 것이다. 그렇지만 정확하게 어디서 얼마 만큼 민감해져야 하는지 아직 제대로 알고있는 사람이 없었다. 이 논문에서는 몇가지 프로그램을 분석하는데 필요한 최소한의 민감성정도를 기계

학습으로 찾아서 보고하고있다. 결과가 놀랍다. 찾고 보니 평균적으로 100개중 2-3개의 함수만 민감하면 (pointer 분석에는) 충분하더라는 것이다. 논문의 주된 내용은 슈퍼컴퓨터로 수십년 돌려야 찾을 수 있는 값을 기계학습을 동원하여 어떻게 빨리 찾는지 테크닉을 다루고 있다. 정작 우리에게 의미있는 것은 결과가 던지는 한줄의 메시지 : 민감해야 하는 정확한 지점을 안다면 많은 민감성이 필요하지 않다.

- **Calling Context Abstraction with Shapes** : 호출문맥을 고려하는 분석분야에서 풀리지 않던 난제가 한가지 있으니, 어떻게 하면 재귀함수를 싹 비용으로 정확하게 분석하느냐는 것이다. 함수호출 문맥을 아무리 자세하게 고려해도 재귀함수를 분석하다 보면 문맥민감성이 부족해서 결국 분석이 부정확해지기 마련이다. 이 논문에서는 함수호출이 만들어내는 call-stack을 재귀적 자료구조로 생각하고 모양분석기법을 적용하여 호출문맥을 요약하는 기법을 제시한다. 실험이 부실하여 실제로 얼마나 분석을 정확하게 만들지는 알 수 없지만, 문맥민감분석이 가야할 길을 한가지 제시했다는 점에서 의미있다고 생각되었다. INRIA 사람들 특유의 이해하기 어려운 슬라이드와 발표자의 프랑스 억양덕에 발표 자체는 따라가기 쉽지가 않았다.
- **Points-To Analysis with Efficient Strong Updates** : 이 논문은 실행흐름을 얼마만큼 존중해야 하는가에 대한 논문이다. 실행흐름을 항상 고려하면 분석 비용이 비싸니 strong update가 발생하는 경우에만 실행흐름을 고려하자는 것이 아이디어이다. 그랬더니 분석비용은 실행흐름을 무시한 분석과 거의 같고 정확도는 실행흐름을 고려한 경우와 거의 같더라는 것이다.

multi-session으로 진행된 학회에는 처음 참석해 보았는데, 참석자로서 준비가 부족한 감이 있다. 최소한 어떤 발표들을 따라서 들을지 미리 고민해보고 스터디를 한번 해 봐야 하는데. 작년 POPL/VMCAI때는 이해가 되지 않더라도 논문을 한번씩 훑어보고 발표를 들어서 꽤 많은 내용을 건질 수 있었는데 올해는 그렇지 못했다. 반성. 또 반성. 아는만큼 얻어간다.