

VMCAI & POPL 2010

Madrid, Spain

2010년 1월 15일 - 25일

[공순호](#), [ROSAEC\(Research on Software Analysis for Error-free Computing\) Center](#)
soon at ropas.snu.ac.kr

요약

2010년 1월 15일부터 25일까지, Spain Madrid에서 열린 POPL'10과 VMCAI'10 학회에 참석했다. 나는 VMCAI'10 학회에 영범형, 왕교수님, 이교수님과 함께 쓴 논문을 발표하였다. POPL'10과 VMCAI'10 학회에 참석하여서 다양한 논문들과 아이디어들을 접하고 우리 분야의 사람들을 만나볼 수 있었다. 이 여행기를 통하여 나의 경험을 통하여서 배운 것들을 다른 사람들과 나누고자 한다.



1. VMCAI'10 논문 발표

VMCAI'10 학회에서 "[Deriving Invariants in Propositional Logic by Algorithmic Learning, Decision Procedures, and Predicate Abstraction.](#)" 논문을 발표하였다. 이 논문은 로파스 연구실의 정영범 박사 과정 학생과 이광근 교수님, 그리고 대만 Academia Sinica의 Bow-Yaw Wang 교수님과 함께 작성한 논문이다. 이 논문은 프로그램 분석의 중요한 문제인 반복문 불변식(Loop Invariant)을 찾는 문제에 대한 알고리즘적인 학습(Algorithmic Learning)을 이용한 새로운 해결책을 제시한 논문이다. 발표에 사용한 슬라이드는 [여기](#)에서 살펴볼 수 있다.

슬라이드를 준비하고, 발표하는 과정에서 배우고 느낀 점은 다음과 같다.

1. 듣는 사람의 관점에서 준비하는 발표 : 듣는 이의 관점에서 발표를 준비할 필요가 있다. 논문을 작성하고, 발표하는 시점이 되면 발표자 본인에게는 논문의 내용이 너무도 익숙하고 당연하게 된다. 하지만 발표를 듣는 청중에게 이 내용은 익숙치 않은 전문적인 내용이 된다. 따라서 이러한 실수는 발표 중간부터 청중을 놓치는 결과를 초래하기 마련이다. 나 역시 이번 논문에 대해서 여러 번 발표하고 후속 논문을 준비하면서 이러한 실수에 빠졌다. 발표를 준비하면서 학주와 원태가 이러한 부분들을 많이 지적해주었고, 그런 의견을 반영하면서 자료를 고쳐나아갈 수

있었다.

2. 두괄식, 두괄식, 두괄식 : 이광근 교수님께서 발표자료를 두괄식으로 만드는 것을 지도해주셨다. 결과적으로 앞부분의 4장의 슬라이드만으로도 전체 시스템을 파악할 수 있도록 구성하였다.
3. 예제 중심의 발표 : 예제 중심의 자료 준비가 발표에 도움이 되었다. 예제를 가지고 설명하는 방식은 듣는 사람에게도 그 내용을 이해시키기가 쉽다. 또한 발표 자체를 부드럽게 하는 효과를 가진다. 이번 경우에도 예행 연습을 하였지만 영어로 발표하는 것이 학회에서는 처음이어서 떨리고 어색했다. 처음의 개괄 부분을 지나고, 우리가 찾아낸 가장 복잡한 예제(parser)에 대한 불변식(invariant)를 사람들이 찾아낼 수 있는지를 물어보고, 답변을 유도해내고, 정답을 알려주는 동안에 긴장이 풀어졌다. 이후 부터는 준비한만큼의 발표를 진행할 수 있었다.



결과적으로 발표는 잘 마무리 되었다. 주된 질문은 불변식을 만들어내기 위한 재료로 주어지는 기초 서술식들(atomic propositions)을 어떻게 만들어 주었는지에 대한 것이었다. 주어진 프로그램의 변수들과 상수들을 이용하여서 만들어낸다고 대답하였다.

2. CAV'10 논문 제출

VMCAI'10 기간과 CAV'10 논문 제출 기간이 겹쳐졌었다. 우리 연구실의 영범형, ROSAEC Center에 방문 연구원으로 왔던 싱가포르 국립대학의 Cristina David, 대만 Academia Sinica의 Bow-Yaw Wang 교수님, 그리고 이광근 교수님과 함께 VMCAI'10의 후속 논문을 썼다. 이 논문은 기존 VMCAI'10 논문이 정량자 없이 표현되는 불변성(quantifier-free invariant)만을 유추해낼 수 있었던 단점을 보완하여서 정렬 알고리즘과 같이 정량자를 포함하는 불변성(quantified invariant)을 유추해내도록 확장한 논문이었다.

열심히 논문을 작성했다. 스페인에 오는 비행기에서, 그리고 도착한 호텔에서 계속 논문 작성에 공을 들였다. Cristina와 Wang 교수님과 시차가 있어서 논문 자체가 24시간 쉬지 않고 변경되었다. 영범형과 이교수님 그리고 내가 논문을 수정하고 잠들면, 그 사이에 Cristina와 Wang 교수님이 논문을 수정하고, 깨어나면 다시 이쪽에서 수정하는 작업이 계속되었다.



논문 자체는 결국 아쉽게도 떨어졌다. 바쁘게 준비하는 동안에 놓치고 지나간 것들이 있었고, 이것들이 결국 문제가 되었다. 하지만 스페인의 호텔에서 논문을 위해서 밤을 지새우며 노력했던 날들이 기억에 남는다.

3. VMCAI'10 & POPL'10 논문들

두 학회에서 발표되었던 논문들 중에서 흥미있던 논문들을 정리해보았다.

"From Program Verification to Program Synthesis." Saurabh Srivastava, Sumit Gulwani, Jeffrey S. Foster

불변식(invariant) 연구에 관련된 논문들을 찾아보면서 저자들의 논문을 읽어왔었다. 이번에 기존의 방법들을 이용하여서 프로그램 분석이 아닌 프로그램을 만들어내는 문제를 푼 논문을 발표한다고 하여서 주의 깊게 들어보았다.

저자들이 기존에 고안했던 프로그램 분석기법을 이용하여서, 프로그램을 만들어내는 문제(program synthesis problem)에 대한 답을 제시한 논문이다. 지난 논문들에서 저자들은 프로그램 분석 문제로부터 제약 조건식들(constraints)을 추출하고 그것을 고정점 연산(fixed-point computation)이나 제약 조건식 풀이기(constraint solver)를 이용하여서 풀어내었다. 이번 연구에서도 동일하게 프로그램을 만들어내는 문제로부터 제약 조건식을 추출해낸다. 하지만 이번에는 미완성된 프로그램 몸체로 인하여 제약 조건식이 불충분하게 추출되게 된다. 이러한 점을 보완하기 위해서 그들은 다음과 같은 추가적인 정보를 필요로 하게 된다.

1. Functional Specifiation : 완성된 프로그램이 만족해야하는 선행조건(precondition)과 후행조건(postcondition)을 명시해 주어야 한다.
2. Domain Constraints: 만들어질 조건식과 프로그램식들이 선택될 영역(domain)에 대해서 명시해 주어야 한다.
3. Resource Constrains
 - a. Control flow template
만들어진 프로그램의 실행 흐름이 어떻게 될지를 명시해 주어야 한다.
 - b. stack template : type -> int
각각의 타입에 대해서, 만들어진 프로그램의 수행과정에서 얼마나 많은 임시 변수들이

사용될 수 있는지에 대하여 명시해 주어야 한다.

c. computation template : op -> int

각각의 연산자에 대해서, 완성된 프로그램의 수행 도중에 얼마나 많이 그 연산이 수행될 수 있는지 명시해 주어야 한다.

결론은 이것이다. "추출된 제약 조건식들과 제공되는 정보들을 풀어내면 프로그램 분석 문제를 풀던 것과 동일하게 프로그램을 만들어내는 문제를 풀 수 있고, 당연하지 않은 프로그램들(non-trivial programs)을 만들어 낼 수 있다."

주어진 문제를 해결하기 위해서 추가적인 조건들이 필수적으로 필요하리라는 것은 예측가능한 것이다. 하지만, 3.b나 3.c와 같은 정보를 입력해 주어야 한다는 것은 들은 적어도 내게는 굉장히 부자연스럽다. 아마도 연구자마다 어떠한 것들이 추가적으로 제공되는 것이 바람직한 것인가라는 질문에 대한 고유의 답을 가지고 있을 것이다. 이러한 것이 이 문제의 공학적인 측면이 아닐까라는 생각된다.



"Continuity Analysis of Programs" Swarat Chaudhuri, Sumit Gulwani, Roberto Lubliner

이 논문은 프로그램의 연속성(continuity)을 분석하는 논문이다. 수학에서의 연속성의 의미는 입력값에서의 작은 차이가 출력에서도 작은 차이로 나타난다는 것이다. 이 논문에서는 그러한 수학적 연속성의 의미를 프로그램 분석에 적용하고 있다.

이러한 것은 프로그램의 안정성(robustness)에 중요하다. 프로그램이 현실세계와 소통하게 되면 그 입력에 작은 오류가 생길 수 있다. 이러한 작은 오류가 프로그램에 미치는 영향은 어떠한가? 만약 그 영향이 크다면 그 프로그램은 현실세계와의 소통에서 문제를 일으킬 가능성이 높다. 따라서 어떤 프로그램이 연속성을 가지는지를 분석하는 것은 의미를 가진다.

논문의 기술적인 측면 보다도, 발상과 동기가 참신하다는 생각을 했다. 모두가 함께 풀었던 문제를 잘 푸는 것이 아니라 남들이 생각하지 않은 하지만 의미있는 문제를 찾아내고, 나름의 풀이를 제공하며, 그 정리를 깔끔하게 해내었기 때문에 우리 분야 최고의 학회에 실릴 수 있는 논문이리라.

논문의 구성에서도, 먼저 반복문(loop)을 포함하지 않는 간단한 경우에 대해서 납득이 가도록 잘 설명한 이후에, 반복문을 추가시키고 그 부분에 대해서 다시 설명하는 방법이 특이하였다. 추후에 논문

을 작성할 때에 배워돌만한 것이라고 생각되었다.

"Automatic Numeric Abstractions for Heap-Manipulating Programs." Stephen Magill, Ming-Hsien Tsai, Peter Lee, Yih-Kuen Tsay

대만 Academia Sinica의 Bow-Yaw Wang 교수님의 소개로 만난 Ming-Hsien Tsai 학생과 Yih-Kuen Tsay 교수님의 논문이다. 새롭게 알게 된 사람들이 발표하는 논문이어서 특히 관심을 가지고 들었는데, 논문 자체의 발상이 새로운 논문이어서 인상적이었다.

문제는 메모리상에 자료구조를 조작하는 프로그램 (heap-manipulating program)에 대하여 정수로 표현되는 그 자료구조의 성질을 검증하는 것이다. 예를 들어서 연결 리스트를 다루는 프로그램이라면 그 리스트의 길이라든지, 혹은 전체 메모리 사용량과 같은 성질들은 정수로 표현될 수 있을 것이다. 이러한 성질을 파악하는 것이 이 논문의 주제이다.



이 논문에서 제시하는 해결책은 메모리 상의 자료 구조를 조작하는 프로그램을 입력으로 받아서 이것을 숫자를 다루는 프로그램(numeric program)으로 변환하고 그 프로그램을 분석하는 것이다. 물론 이 변환이 안전하고 그 분석 결과 역시 원래의 프로그램을 분석했을 때의 결과를 포섭한다는 것을 증명한다.

이러한 아이디어는 어떠한 문제를 이미 잘 풀수 있는 문제로 변환하고 그 문제를 풀어낸다는 점이 흥미롭다. 비슷한 접근 방식은 예전에 학주가 공부해서 발표했던 "Variance Analyses From Invariance Analyses" (by Berdine et.al, presented in POPL'07)라던가, 원태가 작업하고 있는 multi-staged programming analysis를 생각할 수 있다. 이러한 접근 방식은 다른 사람들에게 이해시키기도 좋고, 구현의 일부분을 기존의 것을 가져와서 쓸 수 있는 장점이 있다.

"Shape Analysis of Higher-Order Programs via Abstract Interpretation" Matthew Might

VMCAI'10에서 가장 인상적인 발표였다. Lessig-style을 따라서 큰 글씨 크기의 짧은 문장들로 전체 발표 자료가 구성되어 있었다. 사람들의 주의를 떨어뜨리지 않으면서 전체 내용의 핵심을 정확하게 잘

전달하고 있다는 느낌을 받았다. 이 자료는 web에 [공개](#) 되어 있다.

논문의 내용은 함수를 주고 받는 함수 (higher-order function)에 대한 모양 분석(shape analysis)에 관한 것이다. 모양 분석은 주로 순서 중심의 언어(imperative language)에서 자료 구조를 변경 시키는 프로그램을 분석하는데 사용되어 왔다. Lambda Calculus로 요약될 수 있는 함수를 주고 받는 함수의 경우에는 함수 중심의 언어이기 때문에 한번 만들어진 자료구조가 변화되지 않는다. 따라서 이 발표에서는 먼저 이러한 함수를 주고 받는 함수에서의 모양 분석이라는 것이 A) 어떠한 의미를 갖는지, 그리고 B) 그것이 왜 중요한지를 이야기한다. 그 과정에서 기존의 모양 분석과의 유사관계와 대비를 통하여 설명을 진행하였다.

Utah 대학의 교수로 있는 Matthew Might의 [블로그](#)에는 좋은 글들이 가득하다. 컴퓨터 공학, 특히 프로그래밍 언어를 공부하는 학생이라면 참고할만한 책들과 전공 지식 뿐만이 아니라, 생활 습관들에 관해서도 훌륭한 조언들이 많이 있다. 소중한 글들을 읽으며 즐거웠다.

"Semantics and Algorithms for Data-dependent Grammars." Yitzhak Mandelbaum, Trevor Jim, David Walker

Matt Might의 VMCAI'10 발표와 함께 이번 POPL'10과 VMCAI'10의 가장 훌륭한 발표라고 생각된 논문이다. 간결하게 잘 정리된 슬라이드들이 매우 인상적이었다. 또한 20분의 짧은 시간 동안에 논문의 세부적인 내용에 치우치지 않고, 그 핵심 생각을 전달하고자 노력한 부분이 마음을 사로 잡았다.

내용에 있어서도 훌륭한 논문이다. 먼저 프로그래밍 언어 공동체에서의 생각과 다르게 Parser Generator들이 널리 쓰이고 있지 않다는 현실 인식에서 연구가 출발한다. 적어도 PL 연구자들 사이에서는 이미 끝난 연구라고 생각되는 parsing에 대해서 그 문제가 무엇인지를 조목 조목 나열하고, 현재의 해결책이 어떤 것들이 있었는지를 이야기한다. 이 후에 각각의 해결책들이 조악하다는 것을 설명하고 그들이 새롭게 제시하는 해결책을 이야기하였다.

Trevor Jim이 이야기한 것과 같이 사실 Yacc과 같은 도구를 쓰면서도 불편한 점들이 많았고 이러한 것들을 hacking만으로 해결하는 것에 대해서도 이상하게 생각해본 적이 있었다. 하지만 이러한 것들에 대해서 새로운 해법을 찾으려고 노력하지 않았는데, 이러한 노력을 기울였다는 것 자체로도 좋은 평가를 받기 충분하다는 생각이 들었다. 이교수님이 이야기하시는 우리 주변의 문제, 내가 고통 받는 문제를 해결하는 해결책이야 말로 정말로 좋은 연구라는 것이 이 논문에 해당하는 것이 아닐까? 개인적으로도 Abstract Parsing 때문에 parsing에 관해서 관심이 있어서 더욱 흥미있게 들을 수 있었다.

논문 자체의 내용만으로는 훌륭한 논문들이 너무도 많고, 내가 감히 논할 수 없는 부분들도 많았다. 하지만 순수히 '발표'라는 측면에서는 주관적으로 이 논문이 손꼽히는 논문이라고 할 수 있었다. 발표 자료를 다른 학생들과도 공유하고 싶은데 구할 수 없는 점이 안타깝다.

4. Madrid, Spain

낮 시간에 학회를 마치고 저녁 시간에는 마드리드 시내를 둘러보았다. 인구 300만의 이 도시는 2000년에 방문했었던 바르세로나와는 사뭇 다른 한 나라의 수도에서 느껴지는 묵직함이 있었다. 하지만 서울에 비하여 생각보다 그 중심가가 넓지 않아서 걸어서 2 - 3 시간이면 돌아볼 수가 있었다. Sol역을 중심으로 왕궁과 프라도 미술관이 가장 인상적이었다. 특히 프라도 미술관은 세계 3대 미술관 중의 하나로 손꼽히는 박물관으로 유명한 스페인 화가들(Diego Velázquez, Francisco Goya, El Greco, Jusepe de Ribera)의 작품을 한 곳에서 감상할 수 있었다. 평일 6시 이후에는 입장료를 받지 않아서 부담없이 작품을 감상할 수 있다.



먹을 것에 관한 많은 일들이 기억에 남는다. 초반에 "라도라따"라는 음식점에서 랍스터 100g당 가격을 1마리당 가격으로 오인하여서 다섯 명의 식사 값이 50만원 가까이 나왔던 일은 충격이었고, 마드리드에 대한 첫인상을 좋지 않게 만들어 주었다. 하지만 그 이후로는 특히 조심하여서 그런 불상사는 일어나지 않았다. 마드리드의 모든 식당에는 평일에 점심 메뉴가 있어서 상대적으로 저렴한 가격에 3가지 정도의 음식을 맛볼 수 있었다. Sol 광장 근처에서 맛보았던 하몽(스페인식 햄)과 초콜릿 츠러스가 인상적이었다.

마지막 날에 영범형과 학주, 원태는 톨레도에 놀러를 갔다. 나는 CAV'10 논문 작성에 지쳐서인지 조용히 쉬고 싶었다. 오전을 푹 쉬고 나서 간단하게 호텔 주변과 시내를 걷고 다시 호텔로 돌아왔다. 톨레도에 다녀온 사람들의 사진을 보며 약간 후회가 되기는 했지만 조용히 쉰 하루가 만족스러웠다.