



APLAS/CPP 2013

오학주, 서울대학교

1. 개요

12월 9일부터 13일까지 호주 멜버른에서 열린 APLAS/CPP 2013을 다녀왔다. APLAS (Asian Symposium on Programming Languages and Systems)는 프로그래밍 언어의 전반적인 토픽을 다루는 학회이고 CPP(Certified Programs and Proofs)는 올해로 3년째 된 검증된 프로그램 관련 학회이다. 2011년 첫 공동개최 이후로 총 세번 함께 열리게 되어있었는데 올해가 그 마지막이었다. 기억에 남는 논문발표들을 소개하며 학회기간 동안 있었던 일들을 소개하고자 한다.

2. 논문발표

The Undefined Domain: Precise Relational Information for Entities that Do Not Exist.
Holger Siegel, Bogdan Mihaila, and Axel Simon.

첫날 오후 첫 세션의 의장을 맡았는데, 이 논문이 첫 발표였다. 학회에서 의장을 처음 해보는거라 좀 긴장이 되었다. 오전 세션의 발표들이 대부분 시간을 초과했기 때문에 오후세션에서는 시간을 정확히 지키고 발표끝나기 5분전과 3분전에 싸인을 주기로 했었다. 그런데 이 발표는 너무 빨리 끝났다. 총 25분인데 18분에 끝나서 7분간 질의응답을 해야했다. 그런데 질문이 하나도 없었다. 나라도 질문을 해야하나 하고 생각하는 찰나에 PC chair도 당황했는지 질문을 하나 했다.

발표는 좀 지루했지만, 이 논문의 문제와 아이디어는 재밌었다. 관계분석(relational analysis)를 하다 보면 흔히 겪는 문제인, 두 요약 메모리 상태를 조인할 때 한쪽에서만 정의된 값에 의해 정확도가 떨어지는 문제를 다루었다. 정적 분석에서 이런 일은 매우 흔하게 일어나는데, 예를 들어 함수호출을 문맥을 구분하지 않고 분석할때, 분기문의 합류점을 처리할 때 어느 한쪽 패스에서만 정의된 값들이 자주 등장한다. 값이 한쪽에서는 정의되지 않았다면, 다른쪽에서 어떤 값이 정의되어 있든지 상관없이 정확도를 모두 잃어버리게 된다. 이를 해결하는 가장 일반적인 방법은 context-sensitivity, path-sensitivity등을 높여서 disjunctive 분석을 하는 것이지만 분석의 비용때문에 효과적으로 다루기는 어려운 것이 문제이다.

이 논문의 아이디어는 각 요약 메모리 상태에서 어떤값이 정의되어 있는지를 나타내는 flag 변수들을 도입하고 이 변수들과 일반변수들의 관계를 기저 관계 도메인(underlying relational domain)이 유지하게 해서 조인할때 정확도 손실을 줄이자는 것이다. 패스를 구분하지 않아도, 기존 도메인을 간단히 확장함으로써 염두에 둔 정확도 손실을 줄일 수 있다는 게 논문의 내용이다. 전반적인 아이디어는 신선했지만 논문의 설명방식이 명확하지가 않고 문제를 제대로 잘 푼느낌은 들지 않았다.

Separation Logic Modulo Theory.

Juan Antonio Navarro Perez and Andrey Rybalchenko.

기존의 SMT solver가 separation logic을 지원하도록 하는 방법에 대한 논문이다. 정적 분석에서 theorem prover를 이용하는 경우가 많은데 theorem prover는 또 내부적으로 SMT solver를 이용하는 경우가 많다. 이 때 separation logic을 지원하는 theorem prover를 SMT solver를 이용해서 만들고 싶을 때 어떻게 하면되는지, list에 대한 separation logic에 대해서 설명했다. 핵심은 theorem prover가 SMT solver에게 어떻게 인코딩하여 정보를 줄지, SMT solver의 결과를 해석하고, 만약 반례가 발견되면 theorem prover단에서 반례가 어느부분에 해당하는지 찾는 과정이다. 리스트에 대해서 보였지만 임의의 theory에 대해 적용가능한 방법이란게 장점. 발표가 깔끔하고 이해하기 쉬웠다.

Internal Deployment of the Parfait Static Code Analysis Tool at Oracle

Cristina Cifuentes.

Oracle에서 내부적으로 개발한 정적 분석기인 Parfait에 대한 초청강연. 기존 정적 분석 툴들을 Oracle의 코드에 적용해보니 scalability와 precision에 있어서 모두 만족스럽지 않아서 내부적으로 정적 분석기를 개발해오고 있다고 한다. Coverity사의 Prevent수준으로 메모리 오류들을 찾아내는 것이 목표. 매우 큰 프로그램을 빠르게 분석하기 위해서 간단한 분석부터 시작해서 점진적으로 정확도를 높여가며 필요한 부분만 자세히 분석한다.

회사 기밀때문에 구체적인 얘기는 들을수 없었기 때문에 분석 기술 자체에는 특별한 것이 없어보였지만, 현재 오라클 여러부서에서 성공적으로 쓰이고 있다는 것은 인상적이었다. 아무래도 학계에 있다보면 실제 현장의 문제를 접해볼 기회가 적은데, 그래서인지 발표가 끝나자 사람들이 많은 질문을 했다.

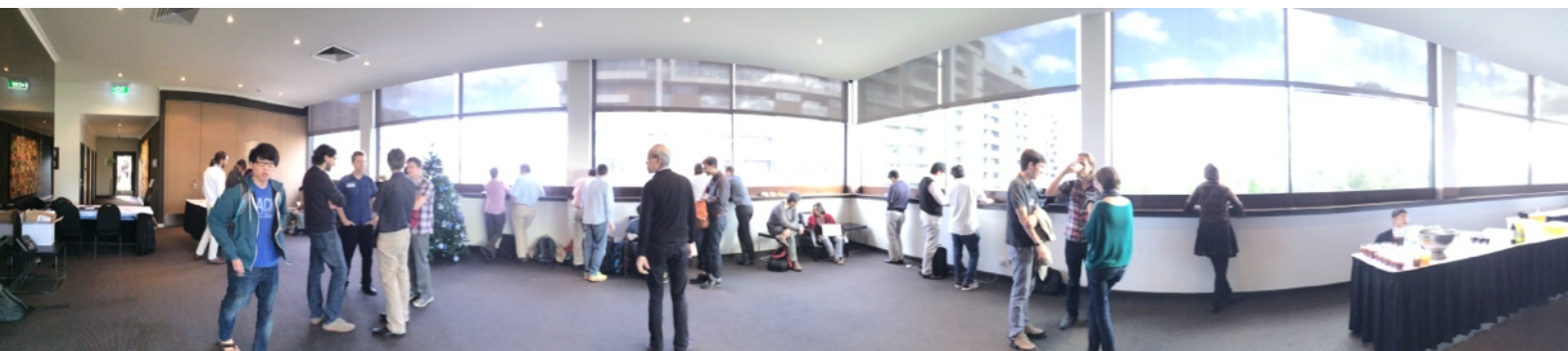
학계와 산업계의 시각 차이도 볼 수 있었다. 처음에 기존 툴들의 허위경보율이 50%나 되기 때문에 쓸수 없다고 하자, Axel Simon이 허위경보율이 50%밖에 안되는데, 두 개중 하나는 진짜 오류란 말인데 좋은거 아니냐고 되묻기도 했다. 생각해보니 두개중 하나가 진짜 오류라면 정말 훌륭한 성능인데, 왜 사람들이 좋아하지 않나 싶었다. 정적 분석이 true alarm을 찾아도 그것은 별로 중요하지 않은 버그인 경우가 많다. dead code에 있는 버그라던지 등등. 찾은 true alarm이 실제로 어떻게 프로그램에 영향을 주게 되는지를 보여줄 수 있다면 false alarm ratio가 좀 높아도 사람들이 좋아하지 않을까 생각이 들었다.

Semantics of Probabilistic Programs: A weak limit approach.

Alessandra Di Pietro and Herbert Wiklicky.

요즘 점점 뜨고 있는 probabilistic programming에 대한 논문. Probabilistic program을 정적 분석하려는데, 기존의 small-step semantics 정의는 정적 분석하기에 좋지 않은 점이 몇 있다고 한다. 그래서 Markov chain을 이용한 새로운 의미구조를 제시한 논문. 직관은 probabilistic program의 의미구조가 현재 상태만 보고 확률적으로 transition이 생성된다는 점에서 markov chain으로 볼 수 있다는 것. 자세한 내용은 이해못했지만 흥미로운 접근 방식임은 분명해 보였다. 이 논문이 전체 논문중 가장 높은 점수를 받았던 것으로 기억한다. 발표 또한 좋았다.

Automated Inference of Library Specifications for Source-Sink Property Verification.



Haiyan Zhu, Thomas Dillig, and Isil Dillig.

이 사람들 연구는 언제나 배경과 문제 설정이 매우 명료하다. 학계내에서 이들의 연구에 대해서 호불호가 갈리는것 같은데, 이 점은 배워야 할 것 같다. source-sink 문제로 표현되는 프로그램 분석이 많은데, 이러한 분석에 대해서 library spec을 추론하는 방법에 대한 것이었다. 해결방법도 직관적으로 쉽게 설명했고 청중의 이해도가 높았던 발표. 아쉬운 점은 실제적으로 정작 중요할것 같은 얘기는 별로 없었고 일반적인 내용만 이야기 하는데 그쳤다는 것이다. 또한 이 논문의 일반적인 틀은 모두 이들의 PLDI'12 논문의 내용과 매우 유사한데도 전혀 다른 테크닉인것처럼 얘기하기도 했다. 하지만 어쨌든 문제를 잘 만들고 깔끔하게 풀어내는데는 능력만큼은 대단하게 느껴졌다.

Fine-grained Function Visibility for Multiple Dispatch with Multiple Inheritance.

Jieung Kim, Sukyoung Ryu, Victor Luchangco, and Guy L. Steel Jr.

카이스트에서 낸 논문이다. 지응이는 이 논문을 발표하기 위해서 뉴헤이븐에서 30시간이상 비행기를 타고 왔다고 한다.

2011년 CPP에서도 지응이의 발표를 들었었는데, 이번 논문은 당시의 언어(FFMM)에 모듈 시스템과 몇가지 함수선언방식을 추가하고 이에대해 타입 안전성을 증명했다고 한다. 즉, 모듈별로 컴파일을 지원하고 함수선언 관련 언어의 표현력을 향상시켰다. 타입 안전성 증명을 위해서 크게 두가지 문제에 대해서 해결책을 제시한다. 첫째는 모듈 시스템을 추가함으로써 발생하는 이름 충돌(name conflicts)을 해결하는 rewriting rule, 둘째는 함수 선언방식을 제한하는 방법이다. 나는 배경지식이 부족하여 자세한 내용은 이해하지 못했지만, 발표는 매끄러웠고 활발한 질문과 토론이 오고 갔다.

류교수님 연구실에서 또 다른 논문(ThisJava: An Extension of Java with Exact Types)도 발표했는데 교수님께서 직접 발표하셨다. 류교수님 연구실에서 자바스크립트 분석기를 개발하면서 실제로 겪은 문제임을 발표 초반에 상세히 말씀하셨는데, 때문에 청중의 몰입도가 높았고 기술적인 내용을 잘 몰라도 무슨 문제를 해결하려고 하는것인지는 확실히 전달이 되었다. 발표후 어떤 사람이 자신도 분석기 코드를 작성하면서 겪었던 문제라면서 흥미로워 하기도 했다.

Proofs, Upside Down (A functional correspondence between natural deduction and the sequent calculus)

Mattias Puech.

재밋고 인상적이었던 발표. Natural Deduction (ND) 스타일로 표현된 논리 시스템을 Sequent Calculus (SC) 스타일의 시스템으로 바꾸는 변환을 제시했다. ND-style로 표현된 논리 시스템은



introduction rule과 elimination rule로 구성되고 각각은 증명에서 아래에서 위 방향과 위에서 아래 방향으로 적용된다. 반면 SC에서는 이러한 양방향성(bidirectionality)이 제거된 형태로 논리 시스템이 표현된다. 이 때 핵심은 elimination rule을 뒤집어서 left rule로 표현하는 것. 이 논문은 이 부분이 함수형 프로그래밍에서 recursive function을 iterative function으로 바꾸는 과정과 유사함을 보이면서, 이를 이용하여 ND를 SC로 바꾸는 변환 룰을 제시했다. 한마디로 “sequent calculus is an accumulator-passing natural deduction” 라고 표현했다. 군더더기 없고 핵심 메시지가 명확하면서 재밌었던 발표였다.

Mostly sound type system improves a foundational program verifier.

Anrew Appel

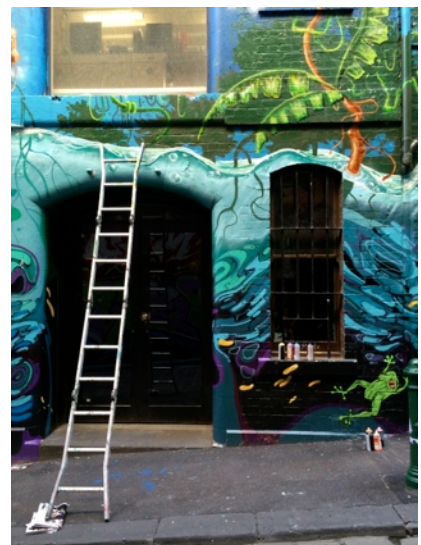
Andrew Appel교수님이 직접 오셔서 발표하셨다. 물흐르듯 부드러웠던 발표. 내용은, 타입 시스템을 이용해서 program logic usability를 높이는 아이디어. C에 대해서 Hoare-style logic을 만들려다 보면 C의 지저분한 면들 때문에 logic자체도 지저분해 진다고 한다. 예를 들어서 $x:=e$ 에 대한 Hoare tripple을 구성할 때 e가 항상 값을 계산해 낸다는 보장이 없으므로 이를 강제하는 side condition이 붙여야 하는 식이다. 문제는 로직이 지저분해지면 이를 사용하기가 번거로울 뿐 아니라 사용자가 side condition을 결국 나중에 다 일일이 확인해야 한다. 이 논문은 정적 분석을 이용해서 이러한 side condition중 일부 혹은 대부분을 자동으로 증명함으로써 logic을 더 편리하게 사용할 수 있다고 보였다. 임의의 정적 분석을 쓸 수 있는데, 이 논문에서는 타입 시스템을 이용한 예를 구체적으로 설명했다.

발표의 말미에서는 구체적인 내용과 배경지식은 이 책에 있다고 하시면서 은근슬쩍 자신의 새로 쓴 책 광고로 넘어갔다. Program logics for certified compilers라는 제목의 책인데 Xavier Leroy와 자신의 제자들과 함께 쓴 이책은 지난 수년간 진행한 연구들을 집대성한 책 같았다. 논문 발표를 하다가 갑자기 책 홍보를 하고 (친절히 ebook은 더 저렴하다는 설명까지) 등등 대가의 연륜과 여유가 함께 전달된 위트 있는 발표였다.

Dynamic Alias Protection with Aliasing Contracts.

Janina Voigt and Alan Mycroft.

Alan Mycroft의 학생으로 보이는 Jania Voigt라는 캠브릿지 학생이 발표했다. 물건중심(object-oriented) 언어에서 aliasing 사용을 제한하는 방식을 제시했다. 언어에서 aliasing을 너무 자유롭게 허용하면 modularity와 encapsulation이 저하될뿐 아니라 고치기 어려운 버그가 발생할 가능성이 높다. 이 논문은 변수를 선언할 때 어떤 object가 read/write 권한이 있는지를 선언해서 런타임에 이를 체크해



주는 시스템을 제시했다. 기존에 이를 위한 다양한 방법들이 있었지만 상대적으로 간단한 디자인과 실제 프로그램에 적용했을때 좋은 성능(오버헤드)을 가짐을 보였다. 논문이 읽기 쉬웠고 깔끔했던 발표.

Practical Alternating Parity Tree Automata Model Checking of Higher-Order Recursion Schemes.

Koichi Fujima, Sohei Ito, and Naoki Kobayashi.

Naoki Kobayashi 연구실에서는 Higher-Order Recursion Scheme (HORS)관련한 연구를 오래 진행하면서 이미 많은 프로그래밍 언어 학회들에서 발표했다. 지금까지 참가한 많은 학회들에서 관련 논문 발표를 들었던 것 같다. 이번에는 HORS에 대한 모델체킹을 처음으로 실용적인 수준으로 적용하는데 성공했다는 발표. 2009년 아플라스에서 Naoki Kobayashi가 invited talk을 할때에는 매우 재미있게 들었던 기억이 있다. 이번에는 학생이 발표했는데, 너무 긴장한 듯 보였고 전달이 잘 되지 않았다.

Model Checking Dynamic Pushdown Networks

Fu song and Tayssir Touili.

발표자의 영어가 유창했음에도 불구하고 전달이 거의 되지 않았던 발표. 청중이 어떤 상태인지는 개의치 않고 25분간 말을 쏟아냈다. 잘 구성이라도 되어 있으면 좋았겠지만, 발표 내용은 거의 논문(주로 definition, lemma, theorem)을 그대로 슬라이드에 옮긴 후 읽어나가는 방식이었다. 시간도 초과되어서 의장이었던 Naoki Kobayashi가 앞에 나가서 시간이 다 되었으니 끝내라는 신호를 수차례 주었는데도 무시하고 계속했다. 결국 강제로 발표를 끝내다시피 하면서 끝났다. 안타까웠던 발표.

3. 사람들

APLAS/CPP 모두 2011년 참가했을때보다 다소 규모가 작았다. 장소가 호주라서 그런것 같았다. 우리나라를 비롯한 아시아권에서도 거리가 짧지는 않은데, 미국이나 유럽에서 온 사람들 얘기를 들어보니 비행기를 20~30시간 타야하는 꽤 험난한 여정이었다. 때문에 미국과 유럽에서는 거의 발표자만 온 듯 했다. 비자문제로 못온 사람들도 있었는데, CPP발표자중 한사람이 비자 문제로 제 시간에 학회에 참석하지 못해서 화상으로 발표를 하기도 했다.

하지만 아담한 규모로 인해 논문발표에 대한 집중도와 사람들간의 교류가 더 활발했던것 같다. 예전에 학회에서 만났던 사람들을 다시 만났으며, 이름은 알고 있었지만 이번에 새로 만나게 된 사람들도 있었다. 먼저, 오랜만에 지응이를 만나서 서로의 연구 등 다양한 얘기를 나눌수 있었다. 카이스트에서는 류석영 교수님께서 성경이와 함께 오셨다. 싱가포르 국립대학의 Chin Wei Ngan 교수님은 학회에서 자주 뵈었는데 요즘 하시는 연구와 크리스티앙과 크리스티나 얘기를 들려주셨다. 닥스틀에서 만났었던 Dillig은 잠깐 인사만 했는데 나중에 얘기를 좀 해보고 싶었으나 발표만 하고 가서 기회가 없었다.

floating point 연산 분석으로 유명한 Eric Gaubault와 Sylvie Putot는 매우 친절한 사람들이었다. 포스터에서 이것저것 물어봤는데 부드럽고 상세한 설명을 해 주었다. Axel Simon을 처음 만났는데 연구 분야가 비슷하고 서로의 논문도 이미 알고 있어서 연구 얘기를 즐겁게 나눌수 있었다. 이 밖에도 캐나다 맥길대학교에서 온 석사과정 학생, 싱가포르 국립대학에서 온 박사과정 학생들과 자주 점심을 같이 먹었다.

4. 멜버른

호주는 우리나라와 적도를 중심으로 열추 대칭적인 위치에 있다. 시차는 2시간밖에 차이나지 않지만, 하루만에 날씨가 겨울에서 여름으로 바뀌었고, 해는 북쪽으로 뜨고, 상현달은 하현달로 보인다. 남반구에 온게 실감났다. 여름이긴해도 멜버른은 아침 저녁으로 쌀쌀했다. 긴팔에 외투까지 걸쳐야 할 정도. 또 날

씨가 시시각각 변해서 하루에 한번은 꼭 비가 오고, 조금 기다리면 날씨가 좋아졌다. 오며가며 들린 시드니와 브리즈번이 화창한 한여름 날씨였던것과 대조적이었다.

멜버른의 거리는 매우 활기찼다. 평일 오후에도 시티내의 레스토랑과 카페는 사람들로 북적거렸다. 알고 보니 멜버른은 먹거리로 세계적으로 유명한 도시였다. 다양한 종류의 사람들(오히려 서양인보다 동양인이 더 많아 보였다)이 모여 사는 곳이다 보니 중국, 일본, 동남아, 이태리, 그리스 등 레스토랑의 종류가 정말 다양했다. 시내 구석구석 골목들을 이들 레스토랑과 카페들이 채우고 있다. 준비한 노천 레스토랑과 카페들이 멜버른 특유의 분위기라 할수 있을것 같다. 한번은 류교수님께서 시내 어느 유명 골목에 위치한 레스토랑에서 켈거루 스테이크를 사주셨는데 분위기와 음식이 모두 일품이었다.

첫 하루 이들은 복잡한 멜버른의 분위기에 적응이 되지 않았었지만 며칠더 머물다보니 호주인들이 왜 멜버른을 가장 살기좋은 곳으로 꼽는지 이해할 수 있을 것 같았다. 차이나 타운이 시티 한가운데 위치해 있을만큼, 다양한 인종과 문화들이 자연스럽게 다채롭게 공존하는 도시였다.

5.마무리

프로그램 위원을 처음 해본거라 더 기억에 남는 학회였던 것 같다. 각 논문들이 생각했던것보다 훨씬 진지하고 까다로운 의견 수렴 과정을 거쳐서 선별되는 과정이 인상적이었고, 내가 리뷰하고 의견을 낸 논문들이 발표되는 것을 보니 보람있었고 저자들과 얘기할때도 더 친근하게 느껴졌다. 또한 학회에서는 대가 교수님들도 발표 준비에 굉장한 노력을 기울이시는것을 보며 보다 더 겸손하고 성실히 연구해야 겠다고 다짐했고, 내 연구분야가 아니더라도 폭넓은 공부를 꾸준히 해야겠다는 자극도 받았다. 학회활동의 재미와 의미를 깨달을 수 있도록 늘 장려 및 지원해 주시는 이광근 교수님께 감사드린다. 또한 편하게 다녀올수 있도록 여러가지로 도와주신 ROSAEC 행정팀 분들에게도 감사드린다.

