

APLAS 2014

Singapore

서울대학교 오학주

1. 소개

11월 17-19 동안 싱가포르 국립대학(NUS)에서 열린 APLAS 2014에 다녀왔다. APLAS는 규모는 다소 작지만 아시아 지역의 연구자들이 한데 모이는 장을 제공해주는 프로그래밍 언어 분야 주요 학회 가운데 하나이다. 올해로 12번째로 열린 아플라스는 한국을 비롯하여 일본, 중국, 호주, 대만, 인도, 싱가포르 등의 나라에서 돌아가면서 열리는데 내년에는 2009년에 이어 다시 한국에서 열리게 된다. 올해에는 유독 일본인들의 참석이 많았다. 논문도 일본에서 가장 많이 냈고 그 뒤로 프랑스, 중국 순이었다. 우리나라에서는 카이스트에서 논문을 한편 발표했고 서울대에서 네편의 포스터 발표를 했다. 3박4일동안의 짧은 일정이었지만 흥미로운 발표와, 반가운 사람들, 맛있는 음식 등 다양하게 학회를 즐길 수 있었다. 재미있게 들었던 발표들과 함께 여행후기를 남긴다.

2. 논문 발표

APLAS는 다루는 주제의 범위가 넓어서 다양한 내용의 논문이 발표된다. 이번에도 프로그래밍 언어 이론, 디자인, 타입 시스템, 계산모델, 컴파일러, 정적 분석 등 매우 다양한 주제의 논문이 발표되었다. 나는 주로 컴파일러, 프로그램 분석, 모델체킹, 보안에 관련된 발표들을 관심있게 들었다. 그 중에서 몇몇을 아래에 소개한다.

Automatic memory management based on program transformation using ownership
Tatsuya Sonobe, Kohei Suenaga and Atsushi Igarashi (Kyoto University)

주제도 흥미롭고 발표도 매우 잘해서 재미있게 들은 발표였다. C와 같이 메모리 관리를 수동으로 하는 언어에서 메모리 해제 명령어(`free`)를 자동으로 넣어서 메모리 누수가 없는 프로그램으로 변환하는 것이 목표이다. 타입기반의 프로그램 변환을 정의하였는데 메모리 안전성을 체크하는 타입 시스템을 정의하였다. 타입을 유추하는 과정을 프로그램 변환으로 정의해서 메모리 해제 명령을 넣을 지점을 찾는다.

이 방식의 안전성과 완전성의 정의가 흥미로웠다. 안전성(*soundness*)는 타입이 맞는 프로그램은 메모리 누수를 일으키지 않는다는 것이고 완전성(*completeness*)은 메모리 해제 명령을 삽입함으로써 메모리 누수 문제가 해결되는 프로그램이라면 제시한 알고리즘은 올바른 위치에 해제 명령을 넣는다는 것이 보장된다는 것이다. 완전성을 보장하기 위해서 불필요하거나 중복되도록 `free`를 넣는 경우가 있을것 같은데도 불구하고 `double free`가 발생하지 않는다는 것도 보장한다고 한다. 실험결과는 없었는데, 실제 프로그램에 대해서 어떤 결과가 나올지 궁금한 연구이다.

Incremental Adoption of Static Typing
Julien Verlaquet (Facebook)

두번째 날 초청강연으로 Facebook에서 정적 타입 시스템을 어떻게 실제 개발환경에 적용해 나가고 있는지 발표하였다. Facebook에서는 주로 PHP를 사용하여 개발하는데, PHP에 정적 타입 시스템을 장착하기 위해서 PHP의 문법과 라이브러리 등을 그대로 유지하면서 `static typing`을 추가한 Hack이란 언어를 새



로 만들었다고 한다. 디자인에 있어서 핵심 목표중 하나는 실시간으로 개발을 하면서 타입 체크를 할 수 있게 하는 것. 이를 위해 PHP함수의 인자와 리턴타입을 명시하도록 한 후 정적 타입 체크를 빠르게 수행한다. 그 밖에도 PHP에는 없는 유용한 라이브러리(Set, Map 등)과 다른 현대 언어들에 있는 기능들이 추가되었다. Facebook에서 약 천만줄의 codebase를 수천명의 소프트웨어 엔지니어들이 공유하며 프로그래밍을 하는데 개발규모와 개발과정의 효율을 위해 새로운 언어를 디자인한다는 것이 인상적이었던 발표.

A method for scalable and precise bug finding using program analysis and model checking Manuel Valiviezo, Cristina Cifuentes and Padmanabhan Krishnan (Oracle Labs)

Oracle에서 자체적으로 개발한 정적 분석기의 성능을 어떻게 향상시켰는지에 대한 논문이다. 주 내용은 프로그램 분석과 모델 체킹에서 쓰여온 다양한 기술들(program slicing, predicate abstraction 등)을 함께 적용하여 정확도와 비용 모두를 만족시키는 실제적인 도구를 만들수 있었다는 것. 기술적으로 특별한 내용은 많지 않았지만 실제 현장에서 유용한 정적 분석 툴을 만들기 위해 그 동안 커뮤니티에서 쌓여온 개별적인 아이디어들을 특정한 형태로 어떻게 결합해야 하는지에 대해서 재미있게 들을 수 있었다. 실험 결과도 인상적이었다. 1.4MLoC의 코드를 분석하는데 1시간 이내에 정확도 80% 수준으로 알람을 내었다. 사실 data flow analysis 기반으로 한다는 이전 버전 분석기도 성능이 놀라웠다. 백만라인 분석하는데 걸리는 시간이 1분이내였고 38개의 알람중 37개가 true alarm이었다.

Model-checking for Android malware detection

Fu Song (Shanghai Key Laboratory of Trustworthy Computing, East China Normal University, China), Tayssir Touili (LIAFA, CNRS & University Paris Diderot, France)

모델체킹으로 악성행동을 하는 안드로이드 앱을 찾는 방법에 대한 논문이다. 임의의 다양한 악성행동을 표현하기 위해서 CTL(Computation Tree Logic)이나 LTL(Linear Temporal Logic)을 이용하고, 안드로이드 앱을 Pushdown system으로 모델링하여 CTL/LTL 모델체킹으로 주어진 앱에 악성행동이 존재하는지를 체크한다. 명세 언어가 충분히 표현력이 있기 때문에 다양한 악성 행동들을 기술할 수 있는 것이 장점이다. information leak뿐만 아니라 picture taking(사용자 몰래 사진을 찍는 행동), video recording(사용자 몰래 비디오를 찍는 행동), 서버로부터 어떤 데이터를 몰래 다운로드 하는 등의 일반적인 malware의 행동을 기술하고 찾을 수 있음을 보여주었다. 실험결과가 좋았는데 1000여개의 안드로이드 앱을 분석하여 기존에는 악성행동이 없는 것으로 알려진 10개의 앱에서 악성행동이 포함되어 있음을 새로 찾아내었다고 한다.

Inferring Grammatical Summaries of String Values

Se-Won Kim, Wooyoung Chin, Jimin Park, Jeongmin Kim and Sukyoung Ryu (KAIST, Korea)

마지막 날 첫번째 발표로 카이스트 김세원 박사님이 문자열 분석에 대해서 발표하셨다. 문법이 있는 문자열을 동적으로 생성하는 프로그램에서, 어떤 문자열들이 만들어질지를 유추하는 정적 분석에 대한 연구이다. 문자열들을 문법규칙의 non-terminal로 요약을 하자는 것이 아이디어의 시작점인데, 이러한 매우 일반적인 아이디어로부터 시작해서 어떻게 실용적인 문자열 분석을 만들 수 있는지에 대한 것이 주요 내용이었다.

연구의 진행 흐름에 따라 어떤 문제가 있었고 어떻게 해결해 나갔는지를 설명하는 스토리있는 발표였다. 따라서 재미도 있었고 따라가기가 수월해서 집중도 잘 되었다. 아이디어 자체도 매우 일반적이어서 이를 구현할 수 있는 다양한 알고리즘들이 제공되면 여러가지 버전의 스트링 분석을 만들어 낼 수 있는 프레임워크가 될 수 있을것 같았다. 이번 발표에서는 CYK파싱 알고리즘을 이용한 스트링 분석만을 소개하였는데, 일반적인 프레임워크와 이것으로부터 다양한 문자열 분석 인스턴스들을 만들어 낼 수 있음을 보인다면 매우 흥미로울것 같았다. 특히 다른 다양한 파싱 알고리즘을 적용하는 방식으로 현재 나와있는 다양한 문자열 분석들, 특히 요약 파싱(abstract parsing)기술이 더 일반적으로 정리될 수 있을지 궁금했다. 이와 관련하여 발표가 끝나고 김세원 박사님과 여러가지 이야기를 나눈것도 기억에 남는다.

Syntax-Directed Divide-and-Conquer Data-Flow Analysis

Shigeyuki Sato and Akimasa Morihata (The University of Electro-Communications, Japan)

컴파일러에서 데이터 흐름 분석(dataflow analysis)의 속도를 빠르게 하기 위해서 분석을 병렬화 시키는 연구이다. 통상적인 데이터 흐름 분석을 하면 분석 알고리즘의 특성상 동시에 수행할 수 있는 작업들이 많지 않아서 효율적인 병렬화가 어려운것으로 알려져있다. 그래서 이 논문에서는 분석을 조립식(compositional)으로 하는 방식을 제시했는데, 조립식으로 하면서 큰 작업들을 동시에 수행하겠다는 것이다. 예를 들면 $if\ e1\ e2\ e3$ 에서 $e2$ 와 $e3$ 의 분석을 동시에 한다든지 하는 셈이다. 눈여겨 볼 만한 부분은 goto 구문이 있어도 조립식 분석이 가능하도록 만든다는 점이다. 그래서 임의의 control flow graph가 주어져 있을때 데이터 흐름 분석을 조립식이 가능하도록 만든다. 이 방식에서는 대부분의 병렬화 작업들이 서로 독립적이므로 스레드수에 비례해서 분석 속도가 향상되었다. 데이터 흐름 분석 뿐만 아니라 좀 더 일반적인 경우에도 이러한 조립식 분석의 아이디어를 적용할 수 있을지 논문을 자세히 볼 필요가 있어 보였다.

3. 싱가포르

싱가폴은 화려하면서도 정돈되어 있는 도시였다. 싱가포르하면 거리에서 쓰레기를 버리면 경찰에 연행된다는 등 다른 나라들에 비해서 법이 엄하고 규제가 많다고 들어왔는데, 역시 입국할 때부터 껌 반입이 금지



되어 있다거나 마약류를 운반하다 잡히면 사형이라고 경고하는 등 다른나라에서 보기 어려운 면모를 보였다. 우리 호텔이 있는 곳은 오차드 거리라는 시내 중심부였는데 거리에서 쓰레기를 찾아보기가 힘들었고 건물들도 마치 모두 새로 지은 듯 깔끔했다. 또한 도시가 전체적으로 화려했는데 특히 밤에는 거리와 심지어 가로수들도 화려한 조명으로 장식되어 있다. 용도가 무엇인지는 불분명하지만 마리나 베이 샌즈 호텔 앞에는 인공으로 만든 정원이 있는데 나무모양의 특이한 건물들과 휘황찬란한 조명들로 환상적인 분위기를 연출하고 있다. 하지만 도시가 너무 인공적이라는 느낌도 들었다. 도시국가이다 보니 자연환경은 없고 대부분의 관광지가 인공적인 것들이다. 너무 화려하게 잘 꾸며놓긴 했지만 그 만큼 금방 싫증날 것 같은 풍경들도 많았다.

음식은 아주 훌륭했다. 적어도 그 동안 가본 학회들 중에서는 가장 만족도가 높았다. 싱가포르 다양한 인종의 사람들이 모여 사는 나라이다 보니 중국, 말레이시아, 인도 음식들이 모두 발달해 있었다. 학회에서는 간식, 점심, 저녁을 모두 제공했는데 이것들만 먹어도 아쉽지 않을만큼 모두 맛있었다. 리셉션은 다른 학회 뱅킷 수준이었고, 뱅킷은 마리나 베이 샌즈 호텔 1층의 해산물 부페에서 열렸다. 마지막 날 출발하기 전에 싱가포르에서 가장 유명한 요리인 칠리크랩을 먹었는데 소스가 특이해서 우리나라 사람들에게도 잘 맞을 것 같았다. 미국이나 유럽에서 열리는 학회들에서는 음식에 대한 기억이 좋았던 적이 별로 없었는데, 아플라스는 2011년 대만에서 열렸을때도 그렇고 음식에 대한 기억이 좋다.

4. 마무리

짧은 일정이었지만 유익하고 즐거웠던 여행이었다. Cristian와 Cristina를 비롯하여 오랜만에 다시 만나게 된 사람들도 반가웠다. 내년에는 포항에서 열리게 되는데 앞으로 우리나라를 포함한 아시아 지역의 PL 연구자들이 더 많아져서 규모도 커지고 질 좋은 논문들도 많이 발표되어서 더 좋은 컨퍼런스로 발전되면 좋겠다. 이번 여행을 지원해주신 이광근 교수님께 감사드리며 여행기를 마친다.

