



PLDI 2012

Beijing, China

2012.06.10 (월) - 2012.06.17 (토)

오학주

서울대학교 프로그래밍 연구실

개요

베이징에서 열린 PLDI 2012에 논문발표를 위해 다녀왔다. PLDI (Programming Language Design and Implementation) 는 프로그래밍 언어 디자인과 구현에 관한 학회로써, 이론의 POPL과 함께 프로그래밍 언어 분야의 대표적인 학회이다. 이번 PLDI는 ECOOP, LCTES, ISMM 등의 국제학회들과 함께 일주일에 걸쳐서 공동 개최되었다. 나는 주로 PLDI와 ECOOP에 참여하였으며, 몇몇 워크샵과 튜토리얼에도 참가하였다.

PLDI 분위기

PLDI는 월,화,수에 걸쳐서 진행되었고 목,금,토에는 ECOOP이 열렸다. 그리고 각종 소규모 컨퍼런스 워크샵등이 동시에 진행되었다. 월,화,수에는 주로 PLDI에 참석했고, 목,금,토에는 ECOOP, SOAP등의 학회, 워크샵, 튜토리얼에 참여하였다.

무엇보다 이번 PLDI에는 한국인 연구자들의 약진이 돋보였다. 우리 연구실에서 국내에서 처음으로 논문을 발표했을 뿐 아니라, 한국인 유학생들이 참여한 논문도 제법 되었다. 우리 연구실에서 2편, 성균관대학교, MIT, 미시간, UC Irvine 등에서 5편으로 총 7편이나 되었고, 전체 48편의 구두발표중에서도 3편을 한국인이 발표하였다. 2008년 PLDI에 참가했을때 한국인 발표자가 한명도 없었고, 참여논문도 거의 없었던 것과 대조되었다. 또한, 가까이 열려서인지 우리나라에서 많은 사람들이 참여했다. 서울대학교, 삼성전자, S-Core, ETRI 등에서 모두 총 28명이 등록했다고 한다.



올해 PLDI발표에는 처음 시도되는 것들이 많았다. 처음으로 아시아에서 열렸을 뿐만 아니라, 광고발표 (teaser talk), 포스터 발표등 논문을 더 알리려는 시도들이 새로웠다. 학회기간(3일)에 비해서 발표 논문수 (48편)가 많았기 때문에 세션을 A,B로 나누어서 동시에 진행되었다. 때문에 참가자는 최대 24편의 구두 발표만 들을 수 있다. 듣고 싶은 발표를 못 듣게 되는 경우를 보

완하기 위해서 발표자가 3가지를 준비해야 했다: 광고 발표, 본 발표, 포스터 발표이다. 먼저, 학회 각 당일 아침에 그 날 발표될 논문이 어떤 내용인지를 각 논문마다 1분씩 광고(teaser talk)를 했다. 그래서 사람들이 어떤 발표를 들을지를 미리 선택할 수 있게하고, 못 듣게되는 논문들의 내용을 대강이나마 알 수 있게 하려는 것이다. 그리고 본 발표가 두 세션으로 나뉘어서 진행되었고, 둘째날 오후에 모든 논문들이 한데 모여서 포스터로 다시 발표되었다. 포스터는 못들었거나 이해하지 못했던 발표를 저자들에게 개인적으로 들을 수 있고, 질문할 수 있었던 시간이라 매우 유익했다. 시간도 충분히 주어져서 거의 모든 논문을 둘러볼 수 있었다. 광고 발표도 취지는 좋았으나 처음 시도되는 것이라 그런지 시행착오가 많아서 원래 의도에 비해서 내용전달이 잘 되

지는 않았다. 발표자로서는 3가지를 준비해야 해서 부담이 더 커지긴 하지만, 전체적으로 발표자나 참가자 모두에게 더 유익한 프로그램이 되었던 것 같다.

프로그래밍 언어 연구가 광범위하게 이루어지고 있다는 인상을 받았다. 고전적인 프로그래밍 언어 주제 이외에도 매우 다양한 분야의 논문들이 발표되었다. 특히, 기존의 문제에 정적 분석등을 활용하거나, 완전히 새로운 시도들도 눈에 띄었다. 대세인 병렬화, JavaScript 등에 대한 연구도 물론 많았다. 기존에 잘 알려진 기술들을 적재적소에 활용하는 시스템을 제시하는 연구가 더 많은 것을 보면서, 이전에 생각하지 못했던 다양한 연구주제들을 경험하였다. 미처 생각하지 못했던 신선한 아이디어와 연구방향을 제시한 논문들도 눈에 띄었다. 그 중 몇가지를 잠시후에 소개하도록 하겠다.

학회 운영에 대해서는 좀 아쉬웠다. 일단 컨퍼런스 호텔이 너무 비싸서 그 옆의 호텔을 예약해야 했다. 그리고 기대했던 벙킷도 실망이었다. 베이징 798 예술특구의 한 전시장에서 열렸는데 좁은 공간에 많은 사람들로 붐볐다. 음식도 변변찮았고 그나마 긴 줄을 서야했다. 벙킷이 일인당 \$100이었는데 그 돈을 다 어디에 쓴 건지 이해하기 어려울 정도였다. 학회 진행도 그리 매끄럽지 않았다. 예를 들어, 2008년에 재밌었던 것 중 하나가 학생연구발표(Student Research Competition)였는데, 이번에는 언제 어디서 하는지 홍보가 제대로 되지 않았다. 처음으로 아시아, 중국에서 열려서 그런지 미국에서 열릴때보다 전반적으로 운영이 미숙했던것 같다.

논문 발표

PLDI에서 스파스 정적 분석(sparse static analysis)에 관한 논문을 발표했다. 그동안 정적 분석 분야에는 요약 해석(abstract interpretation)이라는 강력한 틀이 있어서 정적 분석의 안전성(soundness)을 유지한 채 분석의 정확도(precision)를 자유롭게 조절할 수 있었다. 하지만 정확도를 유지한 채 분석기의 성능(scalability)를 높이는 일반적인 방법은 많지 않다. 우리 논문은 정적분석기 Sparrow의 성능을 향상시켰던 경험을 바탕으로 정확도를 훼손시키지 않은 채 성능을 향상시키는 일반적인 방법을 제시했다. 특히,



특히,

나의 경우에는 박사과정동안 했던 연구의 최종결과에 해당하는 논문으로, 그만큼 애착이 많이갔고 발표도 잘 하고 싶었다.

하지만 발표준비는 리허설을 두번이나 하는 등 어려움이 많았다. 논문의 주 초점이 일반적인 이론을 제시하는 것에 있었기 때문에 어느정도 수식이 들어가는 것을 피할 수

없었다. 직관과 예제로만 설명하다가 자칫 이론의 일반성을 전달하는데 실패할 수 있기 때문이다. 하지만 수식위주로만 설명하면 청중을 잃어버리고 아무것도 전달하지 못할 위험도 있었다. 이 둘의 균형을 맞추기가 어려웠다. 게다가, 기존에 잘 알려진 문제를 푼 것이라기 보다는, 그 동안 사람들이 주먹구구식으로만 늘상 해오던 것을 엄밀하고 정확하게 제대로 정리해보자는 취지의 논문이어서 그러한 취지를 어떻게 설명해야 할지 상당히 많은 고민을 했다. 때문에 3주전부터 슬라이드를 준비하기 시작했지만 떠나는 주가 되어도 슬라이드가 안정화 되지 않았다. 발표에도 자신감이 점점 떨어졌다. 하지만 두번의 리허설을 거치면서 교수님과 동료들로부터 세밀한 코멘트를 받았고, 약간의 수식을 동원하더라도 논문의 핵심 아이디어를 정확하게 전달하는데에 집중하기로 했다. 떠나기 전날에 비로소 슬라이드가 안정화 되었다.



논문 발표는 둘째날 첫번째 세션에 있었다. 발표는 첫째날에 있는게 제일 좋은 것 같다. 그래야 발표가 끝나고 본격적으로 가벼운 마음으로 학회를 즐길 수 있으니 말이다. 역시 첫째날 발표들은 집중해서 듣지 못했고, 특히 오후에는 거의 듣지 못하고 발표 준비를 해야했다. 내용이 입에 붙을때까지 반복해서 연습했고, 연습을 할때마다 점점 자신이 생겼다. 실제 발표는 준비한 그대로 말하고 내려왔는데, 역시 수식이 나오자 사람들이 어려워하는 것이 느껴졌다. 사실 간단한 수식이고 자세히 볼 필요없이 설명의 이해를 위해서만 필요한 정도였음에도 불구하고, 수식 자체에 거부반응을 보이는 사람들이 생각보다 많았던 것 같다. 때문에 처음 도입부에는 집중도가 높았던것 같은데, 기술적인 내용에 들어가면서 집중도가 떨어졌던 것 같다. 또한, 지난 APLAS나 VMCAI때는 발표하면서 대부분의 청중이 한 눈에 들어오는 규모라 교류가 있다는



느낌이 들었는데, 이번에는 규모가 커서 개인이 잘 보이지 않고 청중의 상태가 잘 파악되지 않았다. 질문이 몇개 있었는데 무슨 말인지 잘 이해하지 못했는데, 다행히 오후 포스터 세션에서 질문자와 충분히 이야기할 수 있었다. 발표가 끝난후 그 동안 스파스 분석 관련논문을 통해서 이름만 알고있던 사람들이 찾아와서 서로 궁금했던 것들을 나누는 시간을 가졌다. 발표를 통해서 내용전달을

하는것도 중요하지만, 이를 통해서 관련분야 사람들과 자연스럽게 교류하게 된다는 점도 학회발표의 즐거움 중 하나이다.

발표 직후, 이광근 교수님, 그리고 세션 체어였던 Anders Moller 교수님과 점심을 같이 했다. Moller 교수님은 주로 자바스크립트 분석을 하시는데, 그와 관련해서 문자열 분석, 타입 분석으로 유명한 분이다. 특히 예전의 내 연구와 관련있는 연구도 하신적이 있어서 나도 Moller 교수님 논문을 읽어본적이 있었다. Moller 교수님께서 내 지난 논문들을 유심히 읽어보셨다고 말씀하셨다. 그 동안 논문을 쓰면 누가 읽거나 하는걸까 싶었는데, 반가웠고 더 좋은 연구를 해야겠다고 생각했다.

논문소개

PLDI 논문들 중에서 재미있게 읽은 논문들중 몇 편을 소개한다. 사실 PLDI 발표에서는, 그리 대단해 보이는 발표가 많지 않았다. 그래서 약간 실망하기도 했는데, 한국에 돌아와서 논문들을 다시 읽어본 후 재미있고 좋은 연구들이 많았음을 깨달았다. 짧은 발표시간에 논문의 내용과 의미를 잘 파악하지 못했던 것임을 깨닫고, 미리 논문들의 내용을 파악하는 것을 게을리했던 것을 반성하게 되었다. PLDI외에도 ECOOP과 워크샵들에서 재미있게 들은 발표들도 소개한다.

Automated Error Diagnosis Using Abductive Inference

Isil Dillig¹, Thomas Dillig¹, Alex Aiken²

¹College of William & Mary, ²Stanford University

우리 연구실에서 하는 연구와 가장 관련이 깊은 논문이었다. 이들의 연구는 2008년 PLDI에서 처음 접했는데, 주로 기호실행(symbolic execution) 기반 정적분석기의 성능(정확도, 속도)을 개선하면서 얻은 연구결과들을 논문으로 발표해오고 있다.

이번 논문은 정적분석기 사용자가 겪는 알람분류 문제를 어떻게 도와줄지에 대한 것이었다. 정적 분석기는 보통, 프로그램을 분석 한 후 그 결과인 알람만을 사용자에게 보여준다. 사용자는 각 알람이 진짜인지 가짜인지를 판별하게 작업을 해야하는데, 이것이 사용자에게 정적분석기의 사용에 있어서 가장 큰 걸림돌이 된다. 이 어려움을 덜어주기 위한 이들의 아이디어는 간단하다. 분석기가 알람을 낸다는 것은 해당 성질을 증명하지 못했다는 의미이다. 그 성질을 증명할 정도로 충분한 정보를 분석해 내지 못했더라도 그 때까지 알아낸 정보를 최대한 활용하면 사용자의 알람분류 작업의 노력을 줄일 수 있다는 것이 이 논문의 메시지다. 논문을 읽다보니 왜 이러한 생각을 못했을까 하는 생각이 들었다. 우리는 알람을 보여줄때, 알람정보를 있는 그대로만 “잘” 보여주려고만 했지, 분석기가 이미 해 놓은 작업들을 사용자가 반복하지 않도록 도와주려는 체계적인 생각은 못했던 것이다. 이 논문에서는 *abductive inference*를 이용하여 분석기가 알람을 왜 내게 되었는지, 알람의 근본 원인을 찾는다.

Parallelizing Top-Down Interprocedural Analyses

Awos Albarghouthi¹, Rabul Kumar², Aditya Nori³, Sriram Rajamani³

¹University of Toronto, ²Microsoft Corporation, ³Microsoft Research India

Microsoft Research India에서 발표한 논문이다. `main` 함수부터 시작해서 차례대로 호출되는 함수들을 따라 분석하는 *top-down* 방식의 분석기를 병렬화하는 프레임워크에 대한 내용이다.

*Sparrow*도 *top-down* 방식이라 유심히 들었는데, 우리 방식에는 적용할 수 없고, *demand-driven* 방식의 분석에만 적용할 수 있는 프레임워크였다. 사실 이 논문은 아이디어는 크게 새로울게 없었다. 자연스럽게 병렬로 처리할 수 있는 함수들의 분석을 병렬로 처리하자는 것이 어떻게 보면 전부이다. 하지만 이 아이디어가 잘 동작하는 실제적인 분석기를 이 그룹이 가지고 있었다는 것과 *MapReduce* 개념을 이용하여 일반적인 프레임워크로 구성했다는 점이 인상적이었다. 우리 연구실에서도 여러가지 실험들이 진행되고 좋은 결과도 많이 나오지만 이를 더욱 일반적인 개념으로 설명하고 프레임워크로까지 일반화하려는 노력이 좀 부족했던 것은 아닐까 생각해보게 되었다.

Sound and Precise Analysis of Parallel Programs through Schedule Specialization

Jingyue Wu, Yang Tang, Gang Hu, Heming Cui, Junfeng Yang

Columbia University

병렬 프로그램 분석하는데 있어서 신선한 시도가 기억에 남는 논문이다. 병렬 프로그램을 분석하기 어려운 가장 큰 이유는 각 스레드간에 일어날 수 있는 실행순서(스케줄링)의 경우의 수가 굉장히 방대하기 때문이다. 이 모든 경우를 고려해야 분석이 안전 한데, 모든 경우를 따로 고려할 수는 없고 뭉쳐서 생각해야 하는데 그러다보면 자연스레 정확도는 떨어질 수 밖에 없다. 이 논문은 이 문제를 새로운 시각에서 접근하였다. 먼저 방대한 수의 스케줄링 중에서 몇가지를 선택한 후 프로그램을 그 특정 스케줄링에 대해서 특화(specialization)시킨다. 그러면 그 프로그램은 특정 스케줄링 외에 다른 방대한 스케줄링은 고려하지 않게 되고, 따라서 이 프로그램을 분석하는 것은 모든 스케줄을 고려하면서 분석하는 것보다 훨씬 정확해 진다(실험적으로 포인터 분석의 경우 62%, race detection의 경우 69%의 정확도가 향상됨을 보였다). 그리고 프로그램의 실행중에 고정된 스케줄링에 따라서 프로그램이 실행되도록 강제한다. 혹은 분석된 스케줄이 아닌 다른 스케줄링으로 실행되어야 할 경우에는 동적분석으로 커버한다. 많은 스케줄링중에서 중요하거나 자주 사용되는 스케줄링은 전체의 작은 부분집합이므로 이들에 특화시키고, 특화시킨 스케줄로 실제 실행을 강제해도 문제없다는 아이디어가 신선했다. 논문의 말미에 이 아이디어를 일반적으로 확장하면 직렬 프로그램 분석에서도 많은 경우를 고려하느라 분석의 정확도가 떨어지는 경우에, 이들을 특정 케이스에 고정시켜서 분석할 수 있을 것이라고 써 있는데, 사람들이 기존에 미처 생각해 보지 못했던 새로운 연구방향을 제시한 듯이 보인다.

Self-Stabilizing Java

Yonghun Eom, Brian Demsky

University of California, Irvine

UC Irvine의 엄용훈씨의 연구였다. 발표도 재밌게 들었고, 포스터 세션에서 더욱 자세한 설명을 우리말로 들을 수 있어서 좋았다.

무엇보다 문제자체가 신선하고 재밌었다. 프로그램 실행중에 오류가 발생해도 몇 스텝안에 그 오류가 제거되고 프로그램의 실행이 안정화되는 프로그램들이 있다. 이를 **self-stabilizing** 프로그램이라고 하는데, 이 논문은 이러한 성질을 가지는 프로그램을 판별하는 정적 분석을 고안하는 것이었다. 타입시스템을 가지고 고안되었는데, 크게 두 가지 룰로 구성되어 있다. 목표로 하는 프로그램이 비교적 간단해서 타입시스템을 간단하게 구성해도 해당 성질의 판별이 잘 이루어지는 것 같았다. **self-stabilizing**이라는 개념, 그것을 판별하려는 시도, 제안된 타입시스템 모두 흥미로웠다.

Static Analysis for JavaScript vs. Java

Anders Moller

SOAP (International Workshop on the State Of the Art in Java Program Analysis)의 초청발표였다. 올해 처음생긴 Java 프로그램의 분석에 특화된 소규모 워크샵이다. PLDI가 끝나고 목요일 오전에 ECOOP대신 이 워크샵을 들었는데 마침 첫 시간이 JavaScript분석에 대한 발표였다. Anders Moller 그룹에서 JavaScript분석 프레임워크에 대한 내용이였다. JavaScript가 요즘 가장 이슈화되고 있는 분야임에도 불구하고 이를 분석하는 그룹은 많지 않아보였다. Anders Moller 그룹에서 고안한 JavaScript의 타입분석의 열개를 설명했고, 다른 언어에 비해서 어떤점이 어려운지 보여주었다. 주 이슈는 타입이 동적이라는 것, eval, 함수호출을 정적으로 결정하기 어려운 점 등이였다. 상당히 오랫동안 자바스크립트 분석을 해 온것 같았다. 그래서 관련 논문도 많았고 흥미로웠다. 특히, 예전에 lazy propagation이라는 논문을 재미있게 읽었는데 알고보니 이 아이디어도 자바스크립트 분석을 하면서 나온 것이였다. 발표를 다 들을때쯤엔 JavaScript분석을 해보고 싶은 마음이 일었다.

이 발표가 끝난 후 이어서 Soot Framework위에서 한 분석들에 대한 논문발표가 있었다. 첫 논문은 Soot위에서 RHS 알고리즘을 구현한 내용이였다. RHS알고리즘은 IFDS (Interprocedural, Finite, Distributive, Subset)라고 불리는 분석문제에 대해서 함수 콜/리턴을 매치시키는 함수호출을 고려한 분석을 효율적으로 하는 방법이다 (RHS는 Reps, Horwits, Sagiv 이름을 따서 붙인 이름이다). RHS 논문을 예전에 재미있게 읽었던터라 이 발표도 재미있었고, 발표자가 내용을 쉽고도 명확하게 설명하는 모습이 인상적이였다.

Evaluating the Design of the R Language

Floréal Morandat, Brandon Hill, Leo Osvald, Jan Vitek

Purdue University, USA

ECOOP 논문으로, 내 관심분야는 아니었는데 발표가 훌륭해서 집중하게 되었다. R 언어는 통계 및 그래프를 위한 프로그래밍 언어인데 1990년대 초부터 사용되기 시작했고 지금은 많은 과학자들이 사용하는 언어가 되었다고 한다. R언어가 특이한 점은 굉장히 다양한 프로그래밍 언어 특징들과 패러다임들이 모두 한꺼번에 들어있어 있다는 것이다. 가령, 함수형(functional), 물건중심(object-oriented), 동적(dynamic) 언어의 특징들이 모두 녹아들어있다. 이 논문에서 한 일은 다양한 프로그래밍 언어의 특징

및 기능들 중에서 실제로 사용자가 선택해서 사용하는 것들이 무엇인지에 대해서 조사한 것이었다. 그 결과로 함수형 특징을 사용자가 의외로 많이 사용하고, 기존에 프로그래밍 언어 커뮤니티에서 불필요하다고 여겨졌던 기능들이 실제로는 많이 사용되고 있다는 사실들을 통계적으로 보여주었다. 다양한 패러다임과 기능이 공존할때, 사용자가 선택하는 것들이 무엇인지를 보여주는 재미있는 발표였다.

Yogi: Property Checking via Static Analysis and Testing

Aditya Nori, Sriram Rajamani

Microsoft Research India

Yogi라는 프로그램 검증도구에 대한 튜토리얼이었다. 튜토리얼은 작은 방에 15명정도가 참석하였다. 그 중에서는 Zhendong Su와 같이 유명한 사람들도 꽤 있었으며, 발표자들과 함께 깊이있고 열정적인 토론을 하는 모습이 인상적이었다.

Yogi는 Microsoft Research India에서 2006년경부터 만들기 시작한 검증도구로써 주요 특징은 테스트와 검증을 동시에 활용한다는 점이다. 프로그램 검증에는 크게 두가지 방법이 있다: 버그를 찾는 기법(bug-finding)과 무결점을 검증(verification)하는 것이다. 버그를 찾는데 사용되는 대표적인 기법이 테스트인데, Yogi는 directed testing과 비슷한 테스트 기법을 사용한다. 또 검증기법으로는 SLAM과 같은 모델체커에서 사용하는 기법을 쓴다. 이 두가지를 결합하여 서로 정보를 주고받으면 테스트, 모델체킹을 각각 따로 사용할 때보다 각각의 성능이 더 좋아진다. 예를 들어서, SLAM과 같은 모델체커는 간단한 프로그램을 검증할 때에도 경우에 따라서 많은 시간이 필요할 수 있다. 단순한 요약부터 시작해서 반례가 등장할 때, 딱 그 반례를 제거 할 수 있을만큼 요약을 보정하기 때문이다. 만약, 1000번까지 도는 룩이 있고 그 아래에 검증하고자 하는 성질이 있다면 최악의 경우 1000번의 정교화(abstraction refinement)가 진행되어야 한다. 하지만 테스트는 이러한 사실을 빠르게 알아낼 수 있고, 때문에 테스트 결과로부터 모델체커가 도움을 받을 수 있다. 총 3부로 나뉘어서 진행되었다. 1부에서는 Yogi의 기본 알고리즘에 대한 내용이였다. 포인터와 함수호출이 없는 간단한 프로그램에 대해서 알고리즘이 설명되었고, 다음 시간들에서 함수호출과 포인터를 지원하도록 확장하였다. 그 동안 POPL과 같은 학회들에서 종종 Yogi에 대해서 들어왔는데 이번기회에 전반적인 작동방식에 대해서 이해할 수 있었던 유익한 시간이였다.

베이징, etc

중국이라 역시 음식이 맛있었다. 북경오리와 샤브샤브가 특히 기억에 남는다. 월요일 저녁에 우리 연구실, 중국 NUDT (National University of Defense Technology)의



Liqian Chen, 구글의 하정우 박사님과 함께 베이징 시내 유명한 북경오리집에 갔다. 한 건물이 통째로 음식점이었는데, 층층마다 사람들이 많았다. 겉은 바삭하고 속은 부드러운 오리가 아주 일품이었는데, 오리를 시키면 구운 오리를 직접 테이블 뒤에서 손질해서 바로바로 내주는게 특이했다.

북경오리도 훌륭했지만, 왕푸진 거리에서 먹었던 샤브샤브는 더 인상적이었다. 양고기 샤브샤브 전문점이었다. 일단 우리나라에서 파는 샤브샤브와는 다르게 화로를 따로 주문해야 한다. 우리는 가장 큰 화로를 주문했는데, 왼쪽 사진과 같이 가운데에 숯이 들어있고 가운데에서 물이 솟열로 인해 물이 끓는다. 한국에는 양고기를 파는곳이 일단 많지 않고, 팔더라도 냄새가 난다던지 아주

신선하지는 않은 느낌이었는데, 이곳에서의 양고기는 그런 느낌이 전혀 없었다. 금요일 저녁에 한번 더 방문할 정도로 맛있게 먹었다.

왕푸진은 우리나라 명동 비슷한 곳인데, 작은 골목들에 있는 중국 재래시장으로도 유명하다. 이곳에서는 기상천외함을 넘어서 엽기적인 재료들로 만든 먹거리들을 팔았다. 전갈을 판다는 소문이 있었는데, 전갈은 귀여운 촉에 속했고 매미 유충, 지네, 전갈, 거미까지 실제로 파는 모습에 경악했다.

금요일에는 잠시 학회를 제쳐두고 만리장성에 다녀왔다. 만리장성은 베이징 북쪽에 있는데 버스로 1시간 30분 가량 걸린다. 만리장성은 산세가 꽤 있는 산을 따라 구비구비 끝없이 이어져 있었다. 그 중 아주 일부를 걷는것도 꽤 힘들었는데, 그 옛날 산을 오르내리며 성을 쌓았을 사람들의 수고가 얼마나 대단했는지 상상조차 어려웠다.



맺음말

2008년 PLDI에 처음 참가했을때 학회의 규모와 화려한 논문들 앞에서 위축되었던 기억이 있다. 어설픈 첫 논문을 PLDI에 낸 후 all reject을 받았던 직후라서 더욱 대조되었던 것 같다. 또한 거의 대부분의 논문을 미국대학에서 발표했기 때문에 마치 미국인들만의 잔치에 구경을 왔다는 인상을 받았다. 우리나라에서, 내 연구로 이곳에서 발표하게 될 줄은 상상하기 어려웠다. 오랜기간 지도/지원해 주시고 자신감을 불어넣어주신 이광근 교수님께 진심으로 감사드린다.