Programming Languages and Verification at the University of Colorado Boulder

Pavol Cerny
Bor-Yuh Evan Chang
Sriram Sankaranarayanan
PLV research, right?

Programmers are particular about their language preferences – and most can tell you exactly why they like one over another. Three languages in particular are the rage these days. We compare Ruby, Python, and PHP to see how they stack up.
PLV research at CU has breadth!
PLV research at CU has **breadth**!

How do we **assist reasoning** about programs?

program analysis, developer tools
How do we assist reasoning about programs?
program analysis, developer tools

How do we get reliable, secure software?
verification, model checking
PLV research at CU has **breadth**!

How do we let computers **code for us**?  
* synthesis

How do we assist reasoning about programs?  
* program analysis, developer tools

How do we get **reliable, secure** software?  
* verification, model checking
PLV research at CU has *breadth*!

How do we let computers **code for us**?
- *synthesis*

How do we **assist reasoning** about programs?
- *program analysis, developer tools*

How do we get **reliable, secure** software?
- *verification, model checking*
PLV researchers at CU collaborate!

- synthesis
- program analysis
- verification
PLV researchers at CU collaborate!

concurrency synthesis

You?

program analysis

verification
PLV researchers at CU collaborate!

- **Concurrency Synthesis**
- **Program Analysis**
- **Mobile Security**
- **Verification**

You?
PLV researchers at CU collaborate!

- Concurrency Synthesis
- Program analysis
- Mobile Security
- Transferring Bug Fixes
- Verification

Shawn Meier

You?
How's the debugging going?
PLV is about compassion!
Do you Android?
Do you Android?
Do you Android?
Do you Android?

Crash
I’m not making this up ...
I’m not making this up ...

Android: Crash on rotation, horizontal to vertical
Crash is detected after rotating phone in Gmail Sync now view

[important bug]cordova 1.9 crash on rotation android
5 posts by 2 authors

App crashes when rotating Samsung phone

Issue 20: Crashes when rotating phone horizontally
1 person starred this issue and may be notified of changes.
I’m not making this up ...

Reason:

Out of Memory Error!

App crashes when rotating Samsung phone

Issue 20: Crashes when rotating phone horizontally
1 person starred this issue and may be notified of changes.
State of practice for debugging leaks ...
State of practice for debugging leaks ...

1. Run the app
State of practice for debugging leaks ...

1. Run the app
2. Watch the heap usage
State of practice for debugging leaks ...

1. Run the app
2. Watch the heap usage
3. Dump the heap. Dig around and finally find the culprit!
1. Run the app
2. Watch the heap usage
3. Dump the heap. Dig around and **finally** find the culprit!
State of practice for debugging leaks ...

1. Run the app
2. Watch the heap usage
3. Dump the heap. Dig around and finally find the culprit!
4. Commit a bugfix
State of practice for debugging leaks ...

1. Run the app
2. Watch the heap usage
3. Dump the heap. Dig around and finally find the culprit!
4. Commit a bugfix
5. Bugfix is picked up by our tool Fixr
A Fixr-enabled IDE responds ...
A Fixr-enabled IDE responds ...
A Fixr-enabled IDE responds ...

It looks like you’ve created a memory leak like and 100,000 others. Would you like to apply ?
A Fixr-enabled IDE responds …

It looks like you’ve created a memory leak like and 100,000 others. Would you like to apply ?

the bugfix is “transferred”
Program synthesis for network updates
Program synthesis for network updates

Program synthesis
Program synthesis for network updates

Program synthesis

Network Updates
Program synthesis for network updates

Program synthesis  
Network Updates

Work to appear at PLDI 2015  
First author: Jed McClurg  
(second-year PhD student)
Order Update Example
Order Update Example
Order Update Example
Order Update Example

Property: at all times, maintain H1-H3 connectivity and either traverse A2 or A3
Order Update Example

Property: at all times, maintain H1-H3 connectivity and either traverse A2 or A3

A2-A4-C1 (not good)
A2-A4-T1-C1 ?
Order Update Example

Property: at all times, maintain H1-H3 connectivity and either traverse A2 or A3

A2-A4-C1 (not good)

A2-A4-T1-C1 ?

A2-A4-T1-wait-C1
Artificial Pancreas Verification Project

Collaboration with UC Denver Medical School and UT El Paso.

The diagram illustrates the components and flow of the Artificial Pancreas system. It includes models for Pump Usage, Meal Absorption, Insulin Pump, Artificial Pancreas Controller, Insulin-Glucose Regulatory Model, and Glucose Sensor. The system is designed to verify meal infusion timings and ensure accurate insulin delivery based on meal absorption and plasma glucose levels.
PLV research at CU is **successful**!

**PLDI 2015:** Portland, OR  
McClurg, Hojjat, **Cerny**, Foster. *Efficient Synthesis of Network Updates.*

**PLDI 2014:** Edinburgh, UK  
Logozzo, Fahndrich, Lahiri, **Blackshear**. *Verification Modulo Versions: Towards Usable Verification.*

**POPL 2014 (2):** San Diego, CA  
Coughlin, Chang. *Fissile Type Analysis: Modular Checking of Almost Everywhere Invariants.*  
Jeannet, Schrammel, and **Sankaranarayanan**. *Abstract Acceleration of General Linear Loops.*

**CAV 2014:** Vienna, Austria  
Cox, Chang, **Sankaranarayanan**. *QUICr: A Reusable Library for Parametric Abstraction of Sets and Numbers.*

**ESOP 2015 (2):** London, UK  
Cox, Chang, Rival. *Desynchronized Multi-State Abstractions for Open Programs in Dynamic Languages.*  
Cerny, Henzinger, Kovacs, Radhakrishna, Zwirchmayr. *Segment Abstraction for Worst-Case Execution Time Analysis.*

**EMSOFT 2014 (2):** New Delhi, India  
Ravanbakhsh, **Sankaranarayanan**. *Infinite Horizon Safety Controller Synthesis through Disjunctive Polyhedral Abstract Interpretation.*  
Zutshi, **Sankaranarayanan**, Deshmukh, Kapinski. *Multiple Shooting, CEGAR-based Falsification for Hybrid Systems.*
PLV research at CU is **successful**!

**PLDI 2015:** Portland, OR  
McClurg, Hojjat, Cerny, Foster. *Efficient Synthesis of Network Updates.*

**PLDI 2014:** Edinburgh, UK  

**POPL 2014 (2):** San Diego, CA  
Coughlin, Chang. *Fissile Type Analysis: Modular Checking of Almost Everywhere Invariants.*

**CAV 2014:** Vienna, Austria  
Cox, Chang, Sankaranarayanan. *QUICr: A Reusable Library for Parametric Abstraction of Sets and Numbers.*

**ESOP 2015 (2):** London  
Cox, Chang, Rival. *Desynchronized Multi-State Abstractions for Open Programs in Dynamic Languages.*

**EMSOFT 2014 (2):** New Delhi, India  
Ravanbakhsh, Sankaranarayanan. *Infinite Horizon Safety Controller Synthesis through Disjunctive Polyhedral Abstract Interpretation.*

Zutshi, Sankaranarayanan, Deshmukh, Kapinski. *Multiple Shooting, CEGAR-based Falsification for Hybrid Systems.*
Research at CU is successful!

PLDI 2015: Portland, OR

McClurg, Hojjat, Cerny, Foster. Efficient Synthesis of Network Updates.

PLDI 2014: Edinburgh, UK


POPL 2014 (2): San Diego, CA

Coughlin, Chang. Fissile Type Analysis: Modular Checking of Almost Everywhere Invariants.

POPL 2014 (2): San Diego, CA


CAV 2014: Vienna, Austria

Cox, Chang, Sankaranarayanan. QUICr: A Reusable Library for Parametric Abstraction of Sets and Numbers.

ESOP 2015 (2): London

Cox, Chang, Rival. Desynchronized Multi-State Abstractions for Open Programs in Dynamic Languages.

Cerny, Henzinger, Kovacs, Radhakrishna, Zwirchmayr. Segment Abstraction for Worst-Case Execution Time Analysis.

EMSOFT 2014 (2): New Delhi, India

Ravanbakhsh, Sankaranarayanan. Infinite Horizon Safety Controller Synthesis through Disjunctive Polyhedral Abstract Interpretation.

Zutshi, Sankaranarayanan, Deshmukh, Kapinski. Multiple Shooting, CEGAR-based Falsification for Hybrid Systems.
PLDI 2015: Portland, OR
McClurg, Hojjat, Cerny, Foster. Efficient Synthesis of Network Updates.

PLDI 2014: Edinburgh, UK

POPL 2014 (2): San Diego, CA
Coughlin, Chang. Fissile Type Analysis: Modular Checking of Almost Everywhere Invariants.

POPL 2014 (2): San Diego, CA
Coughlin, Chang. Fissile Type Analysis: Modular Checking of Almost Everywhere Invariants.

POPL 2014 (2): San Diego, CA
Coughlin, Chang. Fissile Type Analysis: Modular Checking of Almost Everywhere Invariants.

CAV 2014: Vienna, Austria
Cox, Chang, Sankaranarayanan. QUICKstart: Support for Symbolic Abstraction of Sets and Numbers.

CAV 2014: Vienna, Austria
Cox, Chang, Sankaranarayanan. QUICKstart: Support for Symbolic Abstraction of Sets and Numbers.

CAV 2014: Vienna, Austria
Cox, Chang, Sankaranarayanan. QUICKstart: Support for Symbolic Abstraction of Sets and Numbers.

ESOP 2015 (2): London, UK
Cox, Chang, Rival. Desynchronized Multi-State Abstractions for Open Programs in Dynamic Languages.
Cerny, Henzinger, Kovacs, Radhakrishna, Zwirchmayr. Segment Abstraction for Worst-Case Execution Time Analysis.

EMSOFT 2014 (2): New Delhi, India
Ravanbakhsh, Sankaranarayanan. Infinite Horizon Safety Controller Synthesis through Disjunctive Polyhedral Abstract Interpretation.
Zutshi, Sankaranarayanan, Deshmukh, Kapinski. Multiple Shooting, CEGAR-based Falsification for Hybrid Systems.
Search at CU is successful!

**PLDI 2015:** Portland, OR  
McClurg, Hojjat, Cerny, Foster. *Efficient Synthesis of Network Updates.*

**PLDI 2014:** Edinburgh, UK  

**POPL 2014 (2):** San Diego, CA  
Coughlin, Chang. *Fissile Type Analysis: Modular Checking of Almost Everywhere Invariants.*

**CAV 2014:** Vienna, Austria  
Cox, Chang, Sankaranarayanan. *QUICr: A Reusable Library for Parametric Abstraction of Sets and Numbers.*

**ESOP 2015 (2):** London  
Cox, Chang, Rival. *Desynchronized Multi-State Abstractions for Open Programs in Dynamic Languages.*

**EMSOFT 2014 (2):** New Delhi, India  
Ravanbakhsh, Sankaranarayanan. *Infinite Horizon Safety Controller Synthesis through Disjunctive Polyhedral Abstract Interpretation.*

Zutshi, Sankaranarayanan, Deshmukh, Kapinski. *Multiple Shooting, CEGAR-based Falsification for Hybrid Systems.*
PLV research at CU has world-wide collaborations!
PLV research at CU has world-wide collaborations!

Collaborators ⇒ Internships and Jobs
PLV students have interned at...

Redmond, Washington
Cambridge, UK
Bangalore, India
The PLV group has fun together!
The PLV group has **fun** together!

Group meetings at the **Boulder Tea House** twice a semester
The PLV group has fun together!

Group meetings at the Boulder Tea House twice a semester

Travel to conferences (POPL 2012)
The PLV group has fun together!

Group meetings at the Boulder Tea House twice a semester

Travel to conferences (POPL 2012)

Our mentoring: Guide you to research that excites you!
Our group

PhD
Shawn  Jed  Alex  Vris

Post-Doc
Peer  Sam  Shadi  Aditya

You?  You?

You?

Faculty

BS
Amin  Evan

Vris

Sriram  Pavol

Max  Evan

Dirk

Sam  Aleks  Hadi  Aditya  Peer
Some of our other research projects

- False alarm triage analysis
- Modular invariant checking
- Analysis of dynamic languages
- Mobile app malware detection
- Incremental verification-validation
- Analysis of medical devices
- Health care process analysis
- Cyber-physical systems verification
- Program synthesis
- Synthesis for software-defined networking

• And soon projects created by you!

http://pl.cs.colorado.edu/