Expeditions in Computer Augmented Program Engineering

http://excape.cis.upenn.edu/

Cornell, Maryland, Michigan, MIT, Penn, Rice, UC Berkeley, UCLA, UIUC

Annual Meeting, June 2013
Software Design Methodology

- **What has changed:**
  - Programming languages
  - Libraries
  - Verification technology

- **What has not changed:**
  - Programming is done by experts
  - Fully specified by conventional programming
  - Verification phase is distinct from design

*Can we leverage modern analysis tools and increased computing power to revolutionize the task of programming?*
Synthesis: A Plausible Solution?

- **Classical:** Mapping a high-level (e.g., logical) specification to an executable implementation
  - Derivation of programs from constructive proofs
  - Synthesis from temporal logic specifications
  - Refinement in model-based design

- **Emerging trends:**
  - Integrating different styles of specifications in a consistent executable (e.g., Program Sketching)
  - Programming by examples (e.g., Flashfill for Excel macros)
  - Programmer interaction and feedback (e.g., Program repair)
ExCAPE Vision

Harnessing computation to transform programming:
Programming made easier, faster, cheaper
Synthesis Tool: Intelligent Assistance

- Designer expresses “what”, possibly using multiple input formats
- Synthesizer discovers new artifacts via integration and completion
- Synthesizer solves computationally demanding problems using advanced analysis tools
- Interactive iterative design
- Integrated formal verification
### Research Organization

<table>
<thead>
<tr>
<th>Tools and Evaluation</th>
<th>Design Methodology</th>
<th>Computational Engines</th>
<th>Challenge Problems</th>
<th>Education and Knowledge Transfer</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Apps for Mobile Platforms</td>
<td>Multicore Protocols</td>
</tr>
</tbody>
</table>

- **Tools and Evaluation**
  - Apps for Mobile Platforms
  - Multicore Protocols
  - Networked Systems
  - Robotic Systems

- **Design Methodology**

- **Computational Engines**

- **Challenge Problems**

- **Education and Knowledge Transfer**
Theme: Computational Engines

- **Reactive Synthesis: From logical specs to finite-state controllers**
  - Compositional synthesis (Vardi)
  - Discrete-event systems theory for concurrency control (Laforetune)
  - Synthesis in presence of identifiers/data (Kress-Gazit, Seshia)
  - Automatic generation of environment assumptions (Alur)

- **Syntax-directed synthesis of code snippets**
  - Modularity for scalability in Sketch (Solar-Lezama)
  - Synthesis from concrete examples + symbolic constraints (Alur, Martin)

- **Hybrid systems: Control of the physical world**
  - Optimal performance of continuous-time controllers (Kress-Gazit)
  - Controllers for linear systems from LTL specs (Tabuada)
  - Theory of robustness for discrete/hybrid systems (Tabuada)
  - Handling nonlinear dynamics for hybrid systems & LTL specs (Kavraki, Vardi)
ExCAPE Specific Activities

- Bridging the gap between research communities in discrete event systems and reactive synthesis (Lafortune, Tripakis, Vardi)
- Application of multiple computational tools in the robotics challenge problem (many PIs)
- Formalization of core computational problem in syntax-directed synthesis of program fragments (Alur, Bodik, Martin, Seshia, Solar-Lezama)
- Syntax-directed synthesis of finite-state controllers for reactive systems (Bodik, Seshia)
Based on input format for SMTLib 2
Problem: Given a formula $\phi$ in an SMT theory with an extra function symbol $f$, and context-free language $L$ for templates, find an expression $e$ in $L$ such that $\phi[f/e]$ is valid.
Theme: Design Methodology

- How best to integrate synthesis in software design practice?

- Rosette (Bodik): Framework to design solver-aided domain-specific languages

- Research on improving scalability and usability of specific tools
  - Sketch for program synthesis (Bodik, Solar-Lezama)
  - LTLMoP for reactive synthesis (Kress-Gazit)
  - TRANSIT for design of distributed protocols (Alur, Martin)
  - Route Shepherd for specifying routing protocols (Loo)
Integrative Design Technologies

- Platform-based design (Sangiovanni-Vincetelli)
  - Framework for representing both high-level algorithms and low-level constraints of implementation platforms
  - Formal support for abstraction, composition, refinement, design exploration
  - Opportunity: Robotics case study

- VELLVM (Martin, Zdancewic)
  - Framework for reasoning about LLVM intermediate program representation and compiler transformations
  - Coq based formalization
  - Opportunity for integrating synthesis tools/algorithms
Challenge Problem: Robot Programming

Goal: Allow end-users to program robotic behaviors

Official Rules

Automatically

(Provably Correct)
Robotic controllers: Research Challenges

- How to consistently integrate physical constraints, sample trajectories, safety rules, and language/temporal-logic requirements?

- How to explain infeasible requirements? How to suggest potential fixes?

- How to program a synthesis engine with completion strategies that take into account the physical and continuous nature of robotics (power, safety, environment traversability)?

- How to address optimality and performance?

- How to evaluate human-robot interaction?

- How to generate control that ports across different robots (different dynamics, control capabilities, safety considerations)?
LTLMoP: Robot control from structured English

Feasible specification

Unsynthesizable specification

Visit all rooms
Research Results

- Improving the scalability of core engine for mapping Temporal Logic formulas to Controllers:
  - Synthesis with identifiers (Kress-Gazit, Seshia)

- Synthesis of cost-optimal plans (Kress-Gazit)

- Motion planning for systems with complex dynamics and LTL specs in partially unknown environments (Kavraki, Kress-Gazit, Vardi)

- Synthesis of controllers with robust performance in presence of uncertainties
  - Theory of robustness for hybrid systems (Tabuada)

- Accuracy in mapping discrete actions to continuous-time trajectories with durations (Kress-Gazit)

- Automatic generation of environment assumptions (Alur, Topcu)
Ongoing Case Study: Robotic Waiter

- **Challenges:** Scalability (items, costumers), uncertainty in sensing and actuation, optimality of behavior, fault recovery
- **Future plans:** exploit symmetries, robust synthesis, task specific abstractions

Initial demo using LTLMoP (Kress-Gazit)
Challenge Problem: Distributed Protocols

- Design challenging due to asynchronous model of communication
- Cache coherence protocols, Distributed coordination algorithms
- Successful application domain for formal verification / model checking
- Correctness involves both safety and liveness properties
- Proposed solution: Allow programmers flexibility

Skeleton based on Extended-Finite-State-Machines

Protocol = + High-level requirements
+ Example behaviors
TRANSIT for Distributed Protocol Design
TRANSIT is a new project influenced by
- Sketch for program synthesis
- Programming by examples

Collaboration opportunities
- Synthesis of distributed algorithms (Alur, Martin, Tripakis)
- Integration with reactive synthesis engines?
- User studies and evaluation?
- Protocol specification using message flows (Talupur, Intel)
- Shared back-end for syntax-directed synthesis
Challenge Problem: Networked Systems

- **Goal**: Automate resource management in networked systems in a safe manner using formal verification and synthesis tools

- **Internet routing protocols**
  - Emerging new platform: Software Defined Networks (SDN)
  - Manual configuration management error-prone

- **Wireless Control Networks**
  - New challenge: Codesign control and routing to ensure stability and performance
Formally Safe Routing (FSR) & RouteShepherd (Loo)

Declarative specifications and constraint solvers used to formally verify errors in routing configurations

Identifying misconfigured routers (in pink) from actual ISP traces.
Control design with aperiodic sampling (Tabuada)
Design of allocation of communication bandwidth and CPU processing over time-triggered network to ensure stability (Alur, Pappas)
Integration of practical controllers (e.g. PID) in WCN (Pappas)
Synthesis for Networked Systems

- Opportunities for increased automation in networked systems management using ExCAPE synthesis solutions

- Mapping virtual network requests to physical network
  SMT-based optimal allocation (Alur, Loo)

- Finding safe sequence of local updates of routing tables
  Computing update sequences using LTL reactive synthesis tools (Loo, Topcu)
Challenge Problem: Programming for Mobile Platforms

- New problem domain as a replacement for “Concurrent Programming”

- Goal: Improve programmer productivity for development of apps
  - Need to adapt to new platforms supported by mobile devices
  - Programmability by end-users can have huge impact

- Ongoing work (Foster and Solar-Lezama)
  Automatic extraction of executable models of Android platform using Sketch synthesis tool
Beyond Four Challenge Problems

- ExCAPE methodology and computational engines has promise in many domains

- High performance computing: Language and compiler for spatial many-core processors (Bodik)

- Synthesis of web-browser user scripts (Bodik)

- Synthesis of GPU layout engines for real-time data visualization (Bodik)

- Synthesis of biological circuits (Bodik)
Theme: Education and Knowledge Transfer

- ExCAPE Summer School: June 13—16, Berkeley; 125 registrants
  Tutorials: Reactive synthesis (Vardi)
  Constraint-based program synthesis (Bodik/Torlak)
  Synthesis for cyber-physical systems (Tabuada)
  + Talks

- ExCAPE Webinar: Monthly talks on diverse topics

- Workshops
  SYNT (at CAV 2013, by Solar-Lezama)
  Synthesis for robotics (at RSS 2013, by Kavraki and Kress-Gazit)
  Special sessions at CDC 2012 & ACC 2013 (by Lafortune)

- Graduate course at Berkeley: Program synthesis for everyone

- Access on online education: MEC in India (Parthasarathy)

- Open source library: OMPL (see ompl.kavrakilab.org, Kavraki)
Synthesis for Online Education

- Emerging opportunity: MOOCs

- Challenge: Personalized feedback on assignments
  - Manual feedback by TAs (not scalable)
  - Grading by peers (not reliable)
  - Evaluation on test cases (how to translate failed tests to errors?)

- Application for ExCAPE tools for synthesis
  - Introductory programming assignments (Solar-Lezama)
  - Modeling and Scheduling problems in Embedded Systems course (Seshia)
  - DFA construction in Theory of Computation (Alur, Hartmann)
    
    see automatatutor.com
Collaboration with Industry & Govt Agencies

- **Industrial Advisory Board**
  - Fix (Intel), Godbole (Honeywell), Godefroid (Microsoft)
  - Gupta (NEC), Kuehlmann (Coverity), Mosterman (Mathworks)
  - Wegman (IBM), Zave (AT&T)

- Research collaborations with industry researchers
  - HP Labs, Intel, Microsoft, Samsung, Mozilla, GreenArrays

- DARPA HACMS program for design of attack-resilient control systems

- iCyPhy center at Berkeley (IBM and United Technologies)

- DOE: Compilers for Exascale machines

- NSF Workshop on Future Directions in Formal Methods (Dec 2012)

- Collaboration with other Expeditions: CMACS (CMU), PPM (MIT)
Theme: Tools and Evaluation

How to integrate many tools being developed by ExCAPE researchers
Sketch, Transit, LTLMoP, ...
and by researchers around the world
RATSY, Comfusy, ...

1. Create a catalog and repository of open-source tools (to-do)

2. Infrastructure to aid design of synthesis tools: Rosette (Bodik)

3. Exchange format to share computational engines and benchmarks
   Syntax-directed synthesis of program snippets (ongoing)
   Reactive synthesis ??
Evaluation

- SynthLib format will help to compare computational capabilities of back-end engines for program synthesis

- Challenge: How to evaluate usability of synthesis tools?

  Proton: Declarative framework for multitouch gestures
  Extensive user study to evaluate effectiveness (Hartmann)

  Ongoing work: Is feedback from AutomataTutor helpful?
  User study planned for Fall semester classes (Alur, Hartmann)
Management and Collaboration

- Challenge: Foster collaboration across disciplines and institutions
- Executive committee: Alur, Bodik, Lafortune, Sangiovanni, Vardi
- Project manager: Liz Ng
- Associate Director search not successful

- Frequent meetings
  - Monthly webinar
  - Face-to-face meeting of all PIs every year
  - Telecons for individual themes/projects
  - Visits by individual PIs to other institutions
Rotating Postdoc Program

- Each ExCAPE postdoc has two mentors, at two different institutions

- Year 2012-13:
  - Ruediger Ehlers (Robotics)
    - Mentors: Kress-Gazit (Cornell), Seshia (UC Berkeley)

- For the upcoming year:
  - Xiaokang Qiu (PhD UIUC), Apps for mobile platforms
    - Mentors: Foster (Maryland), Solar-Lezama (MIT)
  - Indranil Saha (PhD UCLA), Robotics
    - Mentors: Pappas (Penn), Seshia (UC Berkeley)
  - Christos Stergiou (PhD UC Berkeley), Multicore protocols
    - Mentors: Martin (Penn), Tripakis (UC Berkeley)
  - Damian Zufferey (PhD IST Austria), Networked systems
    - Mentors: Loo (Penn), Parthasarathy (UIUC)
Collaboration Success Stories

- Synthesis for programming of robots brings together many PIs ranging from theory to tools, around a common challenge problem

- Bridging the gap between reactive synthesis and supervisory control

- TRANSIT (for distributed protocols) inspired by Sketch

- Standardization of SynthLib for Syntax-directed synthesis

- Automated grading and feedback for online education

- Summer school and emerging community of synthesis researchers