Unified Solver Strategy for Floating-Point based on Proxy Theories

FMCAD 2017 Student Forum

Jaideep Ramachandran
Northeastern University, Boston

Oct 4, 2017
Model Lifting Architecture [FMCAD16]*

f' := f mapped into T

∃σ', σ' ⊨ f'?

no → j' := Refine(f')

yes → σ := (float)σ'

σ ⊨ f?

no → σ := Lift(f', σ', σ)

yes → success

failure

UNSAT

MOLLY = Model Lifting tool

*joint work with Thomas Wahl
# Molly Configurations

<table>
<thead>
<tr>
<th>Spec</th>
<th>Molly&lt;sup&gt;RA&lt;/sup&gt;</th>
<th>Lazy Realizer</th>
<th>Molly&lt;sup&gt;MRFPA&lt;/sup&gt;</th>
<th>Molly&lt;sup&gt;dREAL&lt;/sup&gt;</th>
<th>Molly&lt;sup&gt;RPFPA&lt;/sup&gt;</th>
<th>Approx</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proxy theory</td>
<td>RA</td>
<td>RA</td>
<td>RA</td>
<td>Reals + δ-sat</td>
<td>RPFPA</td>
<td>RPFPA</td>
</tr>
<tr>
<td>Proxy solver</td>
<td>Z3</td>
<td>Realizer++</td>
<td>Realizer++</td>
<td>dReal</td>
<td>Mathsat</td>
<td>Mathsat</td>
</tr>
<tr>
<td>Lifting?</td>
<td>✓</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>×</td>
</tr>
<tr>
<td>Refinement?</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

**Molly** = Model Lifting tool  
**RA** = Real Arithmetic  
**Realizer** = Tool doing eager & exact encoding to Real+Int [DATE14]  
**MRFPA** = Mixed Real–Floating-Point Arithmetic  
**dReal** = Numerical solving tool from CMU  
**RPFPA** = Reduced Precision Floating-Point Arithmetic
- Non-linear polynomials
  \[ 10.25 \leq x^2 + y^2 \leq 10.50 \] 
  - Non-linear complex
    \[ -0.5 \leq e^x + \sin(x) \leq 0.5 \] 
  - Non-linear with operators reordered
    \[ |(x + y)^2 - ((x^2 + (2 \times x) \times y) + y^2)| > \epsilon \] 
  - Linear with operators reordered
    \[ |(x + (y + z)) - ((x + y) + z)| > \epsilon \]

May need different strategies to solve!
Input: \( f \): FPA formula

1. if Linear\((f)\) then
2. \hspace{1cm} return \( \text{MOLLY}^{\text{MRFPFA}}(f) \) \hspace{1cm} \( \triangleright \) mixed real-float reasoning
3. end if
4. \hspace{1cm} result := \( \text{MOLLY}^{\text{RA}}(f) \) \hspace{1cm} \( \triangleright \) pure real abstraction
5. if result \( \neq \) failed then
6. \hspace{1cm} return result
7. end if
8. \hspace{1cm} result := \( \text{MOLLY}^{\text{dREAL}}(f) \) \hspace{1cm} \( \triangleright \) numerical solving
9. if result \( \neq \) failed then
10. \hspace{1cm} return result
11. end if
12. return \( \text{MOLLY}^{\text{RPFPA}} \) \hspace{1cm} \( \triangleright \) reduced precision abstraction
Thank You!