Coverage Metrics for Requirements-Based Testing: Evaluation of Effectiveness

Matt Staats, Michael W. Whalen, Mats Heimdahl
University of Minnesota

Ajitha Rajan
Laboratoire d’Informatique de Grenoble
Testing Process

- Requirements
- Tests
- Talented Developers
- Implementation

Derives
Execute Over

NFM 2010
Testing Process

Requirements → Tests
Tests → Talented Developers
Talented Developers → Implementation
Implementation → Requirements

Derives

Are the tests adequate?

Execute Over

Measure Adequacy Using

White Box Adequacy
Formalized Requirements

“If the onside FD cues are off, the onside FD cues shall be displayed when the AP is engaged”

\[ G(\neg \text{Onside\_FD\_On} \land \neg \text{Is\_AP\_Engaged}) \rightarrow X(\text{Is\_AP\_Engaged} \rightarrow \text{Onside\_FD\_On}) \]

• Possible Coverage Metrics
  
  • Naïve requirements coverage: Single test case that demonstrates that requirement is satisfied
    
    ▪ Prone to “dumb” tests, e.g., execution in which AP is never engaged.

  • More rigorous metrics are necessary
Antecedent Coverage

Many of the requirements in the FGS are of the form:

- Globally if ‘A’ occurs then ‘C’ will occur
  \[ G (A \rightarrow C) \]
- Two ways of satisfying \( (A \rightarrow C) \)
  - A is false
  - A is true and C is true

Antecedent Coverage – Test cases will exercise the antecedent.
Antecedent Coverage

- Many of the requirements in the FGS are of the form:
  - Globally if ‘A’ occurs then ‘C’ will occur
    \[ G (A \rightarrow C) \]
  - Two ways of satisfying \( A \rightarrow C \)
    - A is false
    - A is true and C is true

- Antecedent Coverage — Test cases will exercise the antecedent.

\[ S_0 \rightarrow S_1 \rightarrow \cdots \rightarrow S_n \]

- Not A
- Not A
- A, C

What if:
\[ A \lor B \rightarrow C \]
Unique First Cause (UFC) Coverage

- UFC is an extension of MC/DC to paths
  - Must show individual affect of each atomic condition as Unique First Cause along path
Unique First Cause (UFC) Coverage

- UFC is an extension of MC/DC to paths
  - Must show individual affect of each atomic condition as Unique First Cause along path

Example LTL property - $G( A \lor B \rightarrow C)$. 
Unique First Cause (UFC) Coverage

- UFC is an extension of MC/DC to paths
  - Must show individual affect of each atomic condition as Unique First Cause along path

Example LTL property - \( G( A \lor B \rightarrow C) \).

To show independence of B,
Study Goals

- Is *subsumption* between these metrics indicative of practical effectiveness?
- Are these coverage metrics *any good*?
Case Examples

- Avionics systems courtesy of Rockwell Collins
  - Simulink, translated to Lustre
- Includes “good” set of LTL requirements

<table>
<thead>
<tr>
<th></th>
<th># Simulink Subsystems</th>
<th># Blocks</th>
<th># Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>DWM_1</td>
<td>3,109</td>
<td>11,439</td>
<td>170</td>
</tr>
<tr>
<td>DWM_2</td>
<td>128</td>
<td>429</td>
<td>41</td>
</tr>
<tr>
<td>Vertmax_Batch</td>
<td>396</td>
<td>1,453</td>
<td>294</td>
</tr>
<tr>
<td>Latch_Batch</td>
<td>120</td>
<td>718</td>
<td>110</td>
</tr>
</tbody>
</table>
Basic Experimental Setup

Implementation
- Faulty Versions
- Random Test Sets of Similar Size

Requirements
- UFC Coverage Obligations
- Nonreducible UFC Test Sets

Random Fault Finding %
UFC Fault Finding %
Results

• Does subsumption relate to fault finding effectiveness?
   YES! (Mostly)

• Are these coverage metrics effective measures of adequacy?
   NO!
    ▪ For requirements and antecedent coverage
   YES!
    ▪ For UFC coverage, for 3 of 4 systems
Results

Fault Finding

% Mutants killed

DVM_1  DVM_2  Latctl  Vertmax

Case Example

- Reqs.
- Reqs. Ran
- Ant.
- Ant. Ran.
- UFC
- UFC Ran.
New Questions

• Why do test sets satisfying requirements and antecedent coverage perform poorly relative to random testing?
• Why does UFC's effectiveness as an adequacy measurement vary between systems?
Test Generation Approaches

1. **Requirements**
   - Talented Developers
   - Derives Tests

2. **Random Test Generation**
   - Derives Tests
   - Need more tests!

3. **Measure Coverage**
   - Tests
   - Done!

4. **Automatic Test Generation Tool**
   - Requirements
   - Implementation
   - Done!

---

NFM 2010
Weak Coverage Metrics

- Easy to cheat
- Major problem when using counterexample based test generation
  - Counterexamples intended to be simple traces
  - Simple traces make bad tests
- Counterexample based test generation worst case behavior
  - Positive results are positive
  - Negative results are misleading
- Still, satisfying requirements/antecedent coverage not indicative of good tests
Sensitivity to Requirements Structure

- Problem: UFC based temporal and Boolean operators
- DWM_2 system using many relational and arithmetic operators

Original:

```ltl
LTLSPEC G(var_a > ( 
  case 
    foo : 0 ;
    bar : 1 ;
  esac +
  case 
    baz : 2 ;
    bpr : 3 ;
  esac )
);
```

Revised:

```ltl
LTLSPEC G(var_a > ( 
  case 
    foo & baz : 0 + 2 ;
    foo & bpr : 0 + 3 ;
    bar & baz : 1 + 2 ;
    bar & bpr : 1 + 3 ;
  esac )
);
```
Conclusion

- Evaluated three black box coverage metrics using 4 realistic avionics system
- UFC only useful coverage metric
  - However, UFC is not useful for all combinations of requirements and systems
Questions
UFC Coverage

• $G(A)^+ = \{A \cup (a \land G(A)) \mid a \in A^+\}$
  $G(A)^- = \{A \cup a \mid a \in A^-\}$

• $F(A)^+ = \{\neg A \cup a \mid a \in A^+\}$
  $F(A)^- = \{\neg A \cup (a \land G(\neg A)) \mid a \in A^-\}$

• $(A \cup B)^+ =$
  \[
  \{(A \land \neg B) \cup ((a \land \neg B) \land (A \cup B)) \mid a \in A^+\} \cup \\
  \{(A \land \neg B) \cup b \mid b \in B^+\}
  \]

• $(A \cup B)^- =$
  \[
  \{(A \land \neg B) \cup (a \land \neg B) \mid a \in A^-\} \cup \\
  \{(A \land \neg B) \cup (b \land \neg (A \cup B)) \mid b \in B^-\}
  \]

• $X(A)^+ = \{X(a) \mid a \in A^+\}$
  $X(A)^- = \{X(a) \mid a \in A^-\}$