Conformance Testing for Interoperability of Personal Healthcare Devices

Jeff Lei, UT Arlington
ylei@cse.uta.edu
For NIST Health IT Symposium
Sept. 26, 2013
Acknowledgement

• Our ProTest Team
  – UTA: Linbin Yu, Feng Duan
  – NIST: Raghu Kacker, Rick Kuhn, Ram D. Sriram, and Kevin Brady.

• We are grateful for the support from ITL!
  – Grant #: 60NANB12D021
Outline

• What is IEEE 11073 PHD?
• What is Conformance Testing?
• What is T-Way Sequence Testing?
• ProTest: A Prototype Tool (Demo)
• Conclusion and Future Work
IEEE 11073 PHD

• A family of standards that allow personal healthcare devices to interoperate with each other.

• Optimized for the unique characteristics of personal healthcare devices
  – Portable, energy constrained, and limited computing capacity

• Promoted by Continua Health Alliance (http://www.continuaalliance.org/)
  – More than 200 member companies, including IBM, Intel, Cisco, Philips, Samsung, and others.
Agent and Manager

• **Agent**: A device used to obtain measured health data from the user.
  – blood pressure monitors, weighing scales, blood glucose monitors, and others

• **Manager**: Manage and process data collected by one or more agents.
  – personal computers, smart phones, set top boxes

• Manager devices are typically less powered constrained and have more computing capacity.
A Typical Setup

Agents

IEEE 11073 PHD

Managers

Remote Services

- Health Monitoring
- Fitness Advising
- Diet Advising
- Aging Service
IEEE 11073-20601

- A core component of 11073 that defines rules for data exchange between an agent and a manager.
- Defined at the application layer and can work with different transport protocols
  - Bluetooth, USB, ZigBee, and others.
The Agent State Machine

From 11073 Specification
The Manager State Machine

From 11073 Specification
An Example Scenario

- Association request
- Association response (accepted)
- Confirmed Event Report (weight)
- Acknowledgement (OK)
- Association Release Request
- Association Release Response

Check system ID and Config ID

Process event report

Prepare to release
Outline

• What is IEEE 11073 PHD?
• What is Conformance Testing?
• What is T-Way Sequence Testing?
• ProTest: A Prototype Tool (Demo)
• Conclusion and Future Work
Conformance Testing

- In order for agents and managers to interoperate each other, they must implement IEEE 11073 correctly.

- **Conformance testing** is to test agent or manager implementations to ensure that they conform to their protocol specifications.
Testing in General

• Three major steps
  – Test generation, test execution, and test evaluation
• Often impractical to test all possible scenarios
  – What scenarios to test? When to stop?
• The key is to be systematic, i.e., follow a well-defined strategy
  – The notion of coverage is often used to ensure test adequacy
• Testing can easily take more than half of the development budget
  • This is specially so in the medical domain!
Conformance Testing in Focus

- Typically a black-box, model-based approach
  - Does not require access to source code
  - Tests are generated from a model, i.e., specification, instead of the implementation

- Multiple levels of conformance testing
  - **Message level**: Ensure syntactic and semantic conformance of individual messages
  - **Sequence level**: Ensure conformance for sequences of message exchanges.
Conformance vs Interoperability

- **Conformance testing** typically tests individual implementations against their specifications,
- **Interoperability testing** actually puts multiple implementations together to see if they could interoperate with each other.
- Conformance testing can significantly increase the likelihood of interoperability
State of the Art

• **Automata-Theoretic Methods**: Generate test sequences to guarantee detection of certain types of errors
  – Missing transitions/states, incorrect transitions, output errors, and others
  – Impose certain assumptions and often require a large number of test sequences
  – Examples: W-method, Wp-method, UIO-method, and others
Coverage-Based Methods: Generate test sequences to achieve a coverage goal

- State cover, transition cover, boundary-interior cover, and others
- No guarantee on fault detection, but more practical in terms of assumptions and number of test sequences
Our Approach

• A coverage-based method that applies t-way testing to conformance testing of medical devices.

• T-way testing has been shown very effective for general software testing.

• Our initial results suggest that t-way testing has the promise to significantly increase the quality of conformance testing while cutting its cost.
Conformance Testing is a major component in the development of the core health IT Testing infrastructure led by NIST.

Specifically, our work is in the area of Conformance Testing of Medical Devices.

- NIST tools, ICSGenerator and ValidatePDU, seem to work at the message level.
- Our work complements them at the sequence level.
Outline

- What is IEEE 11073 PHD?
- What is Conformance Testing?
- What is T-Way Sequence Testing?
- ProTest: A Prototype Tool (Demo)
- Conclusion and Future Work
The Test Explosion Problem

• Testing is one of the most widely used approaches to ensure software quality.

• However, the number of possible tests is often huge (and even infinite)
  – About 10 million possible tests for a system with 10 5-value parameters.

• **Challenge**: How to select a small number of tests that are effective for fault detection?
A Bug’s Perspective

• As a whole, the behavior of a system could be affected by many factors.

• However, individual bugs are often affected by only a few factors.
  – A widely-cited NIST study suggests no more than 6 factors for practical applications.

• But, we do not know “what” parameters affect “what” bugs.
The NIST Study

(Kuhn, Wallace, Gallo, 2004)
T-Way Testing

• A t-way test set covers all the t-way combinations, instead of all possible combinations (of all the parameters)
  – No need to know “what” parameters cause “what” faults.

• Extremely effective yet substantially reduces the number of tests
  – 10 5-value parameters (about 10M possible tests):
    2-way testing – 49 tests; 3-way testing – 307 tests; 4-way testing – 1865 tests
Consider a system that has three parameters, each having two values 0 and 1.

Pick ANY two parameters, all combinations 00, 01, 10, 11 are covered.
T-Way Sequence Testing

• Expands the domain of t-way testing from test data generation to test sequence generation
  – Order must be taken into account

• Many programs exhibit sequence-related behavior
  – Web applications, multithreaded programs, network protocols, and others

• Key Idea: Instead of testing all possible sequences of all the events, we only test all possible sequences of any t events.
• Every t-way target sequence must be covered by at least one test sequence
  – A t-way target sequence is a sequence of t events that can be executed in the given order
  – A test sequence covers a target sequence if the t events in the target sequence are executed in the same order
T-Way Sequence Generation

a. Compute all the t-way target sequence
b. Build a shortest path $P$ to cover a t-way target sequence
c. Extend $P$ to cover as many target sequences as possible
d. Remove all the target sequences that are covered by $P$
e. Repeat steps b, c, and d until all target sequences are covered
1. All 2-way target sequences:
   - ab, ac, ad, bb, bc, bd, cb, cc, cd
2. Start from S0->S1->S3 (ab), append S4: S0->S1->S3->S4
3. Build test sequence a->b->d which covers ab, ad, bd
4. Remaining 6 targets:
   ac, bb, bc, cb, cc, cd
5. Start from S0->S1->S2 (ac), build test sequence a->c->b->b->d and covers ac, cb, cd, bb, bd
6. Build a->c->b->c which covers cc
7. All targets are covered.
T-Way vs Transition Cover

• Transition cover requires every transition to be tested once
  – No attempt made to test interactions of events
• Problems due to interaction of events may not be detected by transition cover
  – For example, assume that event a affects event b, and something goes wrong with a.
  – This problem will not be detected if a and b are tested in different sequences.
• T-way testing guarantees detection of such problems!
Long vs Short Sequences

• Longer sequences can reduce the total test length as well as start-up and tear-down cost.
• However, it is often difficult to debug a long sequence if a failure is detected.
• In our approach, the length of a test sequence is restricted by allowing the same event to occur for no more than $t$ times.
Outline

• What is IEEE 11073 PHD?
• What is Conformance Testing?
• What is T-Way Sequence Testing?
• ProTest: A Prototype Tool (Demo)
• Conclusion and Future Work
ProTest: A Prototype Tool

• Streamlines the entire testing process, and also integrates with a tool, i.e., LCOV, to collect code coverage
• Supports both transition cover and 2-way sequence testing
• Provides a GUI that allows one to easily operate, visualize and inspect the execution of test sequences
• Written in Java and thus runs on different platforms
The Framework

- Generate a set of test sequences that achieves edge or 2-way coverage based on the state machine model
- For each test sequence, generate test data that are needed to execute the sequence
  - For example, an EVENT REPORT event in a test sequence must be populated with actual report data
- Automatically execute each test sequence and then evaluate the test result
The Framework (2)
Antidote

• An open-source C implementation of 11073-20601’s protocol stack
  – Mainly tested on Linux, and also has a port for Android.
  – Developed by Signove, a Connected Health company in Brazil

• Used by ProTest as the System Under Test
Other x73 Implementations

- **EtherMind** by Mindtree (commercial, ANSI C)

- **OpenHealth** by LibreSoft (open source, Java)
  - Mainly developed for Android
  - [http://openhealth.libresoft.es/node/45](http://openhealth.libresoft.es/node/45)

- **Medical Connectivity Library** by Freescale
  - (free with Freescale processors, ANSI C)
Initial Results

• About 75% of statement coverage in the communications package
  – No attempt to try different message types, some transitions are not implemented

• Detected two bugs of Antidote
  – One confirmed by an Antidote developer, and the other was fixed in the latest, but not released, version

• Demonstrated that 2-way sequence testing can detect bugs that cannot be detected by state cover testing.
The Two Real Bugs

• Transition mismatch
  – A transition labeled by event rx_aarq was defined for state checking_config
  – However, in the actual code, three transitions were implemented for three sub-types of event rx_aarq_*, which can never be fired.

• Invalid message construction
  – The length of a message rx_roer was computed incorrectly, which results in a rejection by the encoding module.
Consider the following event sequence:

- Agent sends an unknown but acceptable configuration
- Manager asks for more information and then registers this configuration
- Agent sends the same configuration again
- Manager recognizes this configuration as a known configuration

What if there is a bug such that Manager does not register the configuration correctly?
Outline

• What is IEEE 11073 PHD?
• What is Conformance Testing?
• What is T-Way Sequence Testing?
• ProTest: A Prototype Tool (Demo)
• Conclusion and Future Work
Current Status


• Built a project website that collects the relevant resources
  – http://barbie.uta.edu/~lyu/healthcare/

• Built a prototype tool that automates the testing process, but only for the Manager side.

• Conducted an initial study on the effectiveness of our approach.
Next Steps

• Extend ProTest to cover the Agent side
• Apply t-way testing to generate message data
  – Currently message data are generated to allow each test sequence to be executed once (i.e., no data coverage is achieved.)
• Develop a framework to allow more complex test evaluation rules
  – For example, the user may add evaluation rules based on their experience
  – Currently, we only check the response type.
Next Steps (2)

• Conduct empirical studies to evaluate our approach
  – Real-life and seeded faults, comparison with other methods, other open-source 11073 implementations

• Generalize ProTest for testing other healthcare protocols
  – Separate protocol-independent part from protocol-specific part, and provide a well-defined interface for the protocol-specific part.
Next Steps (3)

- Develop a comprehensive methodology and toolset for t-way conformance testing of healthcare protocols
  - Test data/sequence generation, and testing individual messages as well as sequence of message exchanges
Personal health devices play an increasingly important role in the healthcare solutions.

Our initial results indicate that t-way testing can be very effective for testing healthcare protocols.

Our vision is to develop a comprehensive set of t-way testing methods and tools for conformance testing of healthcare protocols.

- Consistent with NIST’s Healthcare Initiative!
Questions?