Logic Model Checking of the Delay Tolerant Networking’s Bundling Protocol

Ed Gamble

Laboratory for Reliable Software
Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA

2010 Workshop on Spacecraft Flight Software
8 December 2010

NASA ESMD (ESAS/6G) Intelligent Software Design Project
Outline

- Context
- DTN Bundling Protocol Overview
- BP Protocol Verification
- ION Implementation Verification
- Conclusions
Context

- Interoperability between N subsystems, based on a defined interface protocol, may require as many as \( N^2 \) pairwise verifications.
  - Can logic model checking be applied effectively?
  - Does logic model checking offer advantages beyond what a comprehensive protocol test suite would?
- Task: Identify a protocol with multiple implementations; build and apply a logic model
DTN Bundling Protocol

- The DTN Bundling Protocol reliably routes and forwards ‘application data units’ between defined endpoints across a delay tolerant network.

- Bundling Protocol Characteristics:
  - **Well-Defined** - based on RFC 5050
  - **Multiple Implementations** - Notably ION, at JPL
  - **Small Size** - logic model is computationally feasible
  - **Expert Consultant** - BP co-author at JPL
DTN Bundling Protocol

- Key Capabilities:
  - custody-based retransmission
  - cope with intermittent connectivity
  - exploit scheduled, predicted and opportunistic connectivity
  - late binding of overlay network endpoints.

- Logic model focuses on the first two
DTN Bundling Protocol
DTN Bundling Protocol

- Correctness Claims
  - **ADU Receipt w/o Errors**: In the absence of errors, an ADU that is transmitted (by the source EID) must always be received (by the target EID).
  - **ADU Accountability**: An ADU transmitted with a request for one or more types of ‘bundle status reports’ shall eventually receive each and every one of the requested status reports.
BP Logic Model Architecture

[Diagram of the BP Logic Model Architecture]

SPIN Verifier

- App #1
- Bundle Node #1
- Bundle Node #2
- Bundle Node #3
- Bundle Node #4
- App #2

Connections:
- AA between App #1 and Bundle Node #1
- CLA between Bundle Node #1 and Bundle Node #2
- CLA between Bundle Node #2 and Bundle Node #3
- CLA between Bundle Node #3 and Bundle Node #4
- CLA between Bundle Node #4 and App #2
- AA between App #2 and Bundle Node #4
BP Logic Model Results

- Both correctness claims were partially verified.
- The SPIN verification runs did not fail but could not run to completion - owing to computational resource constraints.
- SWARM technology applied; no errors identified.
ION - A BP Implementation

- ION - Interplanetary Overlay Network - Designed specifically for interplanetary communication:
  - Slow, asymmetric, over-subscribed links
  - Limited computational resources
  - High reliability; predictable performance
ION Logic Model

- Primary Constraint: The internal state of a networking node (ION implementing BP) is not accessible to the SPIN verifier.
  - Implications:
    - SPIN cannot backtrack through ION’s state
    - The BP behavior must appear to be atomic.
ION Logic Model Architecture

SPIN Verifier

Driver

- Randomize
- Send
- Recv*
- Confirm*

Bundle Node

- SPIN to ION
- CLA
- ION to SPIN

ION

BP Nodes
ION Logic Model Results

- For the two BP correctness claims:
  - ADU Receipt w/o Errors: Pending
  - ADU Accountability: Fully verified
Conclusions

- A viable N subsystem intercommunication verification process is:
  - Identify a protocol; identify correctness claims
  - Develop a ‘comprehensive test case’ logic modeling architecture supporting fully randomized test state.
  - Test the protocol (if modeled); test implementations.