Distributed Protocol Conformance Testing

Prof Pallapa. Venkataram,
Department of Electrical Communication Engineering
Indian Institute of Science
Bangalore – 560012, India
Distributed architecture by local methods

General distributed test architecture
FSM representation of IUT with n ports

- A multiport FSM with n ports (np-FSM) - 6-tuple \( (S_t, \Sigma, \nu, T, O, S_0) \) where \( n > 1 \)

- \( S_t \) is finite set of states or labels,

- \( \Sigma \) is a n-tuple \( L_{i_k} = (L_{i_1}; L_{i_2}; \ldots; L_{i_n}) \), where \( L_{i_k} \) is set of inputs for port \( k \)

- \( \nu \) is a n-tuple \( L_{o_k} = (L_{o_1}; L_{o_2}; \ldots; L_{o_n}) \), where \( L_{o_k} \) is set of outputs for port \( k \)

- \( T \) is a transition function over subset \( D \)

- \( O \) is an output function, that maps \( D \) to \( L_0 U \{ \not\in \} \)

- \( S_0 \) is the initial state belongs to \( S_t \).
Functioning of 3p-FSM

1. \( t_1 = 1 = < a; c > \)

2. \( t_2 = 2 = < d > \)

3. \( t_3 = 3 = < d > \)

Execution of transitions are as follows:

- \( t_1: A \text{ to } B = 1/< a; c >, \)
- \( t_2: B \text{ to } C = 2/< d >, \)
- \( t_3: C \text{ to } A = 3/< d >, \)
- \( t_4: A \text{ to } A = 3/< c >, \)
- \( t_5: A \text{ to } B = 2/< a; d >, \)
- \( t_6: B \text{ to } B = 1/< d >, \)
- \( t_7: B \text{ to } C = 3/< a; c >, \)
- \( t_8: C \text{ to } C = 2/< a >, \)
- \( t_9: C \text{ to } A = 1/< a >. \)
Functioning of 3p-FSM

The initial state is A

Li₁ = {1},
Lo₁ = {a, b},

Li₂ = {2},
Lo₂ = {c},

Li₃ = {3},
Lo₃ = {d},

Port 1

Tester 1

Port 2

Tester 2

Port 3

Tester 3

A

B

C

t1: 1/<a,c>
t2: 2/<d>
t3: 3/<d>
t4: 3/<c>
t5: 2/<a,d>
t6: 1/<d>
t7: 3/<a,c>
t8: 2/<a>
t9: 1/<a>
Synchronizable Test Sequence

− Interaction Points

• Interaction points (IPs) of a given transition \([pi; PO]\). Let \(pi \in \{1; 2; n\}\), and PO is a subset of \{1; 2; ...n\} (n is number of ports). \([pi; PO]\) is said to be an interaction point of a given transition \(t\) if \(t\) receives an input from the port \(pi\), and sends output(s) on port \(PO\) (if \(PO=\)null, \(t\) does not send any output).

− Synchronizable test sequences: Given an ordered pair of transitions \(t1\) and \(t2\) of machine, let \([pi,PO]\) and \([pi', PO]\) be their IPS, respectively. Then \(t1\) and \(t2\) are asid to be synchronizable if \(pi=pi'\) or \(pi'\) PO.

− A given test sequence is said to be synchronizable if any two consecutive transitions of the sequence are synchronizable.

− As an example for the 3p-FSM, the Ips the transitions \(t1\) and \(t2\) are \([1,\{1,2\}]\) and \([2,\{3\}]\) respectively and transistions \(t1\) and \(t2\) are synchronizable.
- Synchronization problem:

  - Considering two consecutive transitions $t_1$ and $t_2$ of a given np-FSM ($n \geq 2$), one of the testers is said to face a synchronization problem if the tester did not take part in the first transition and if the second transition requires that it sends a message to the machine I.

- For example, consider 3p-FSM shown in the figure, Tester 3 faces a synchronization problem when considering the consecutive transitions $t_1$ and $t_7$.

- Tester 3 supposed to send the input 3 to the machine after the $t_1$ has been to the tester 3, tester 3 does not know whether $t_1$ has been executed.

- So the derived test sequence must eliminate this synchronization problem.
Synchronizable Test Sequences

Given a ordered pair of transitions $t_1$ and $t_2$ of machine $I$, let $[p_i, P_o]$ and $[p'_i, P'_o]$ be their IPs, respectively. $t_1$ and $t_2$ are said to be synchronizable if $p_i = p'_i$ or $P_o = P'_o$.

- **2p-FSM and its synchronizable transition tour**

- **Input Sequence:** $0, 1, 1, 0$
- **Port 1:** $\begin{array}{c} 0 \\ b \\ a \\ 1 \\ c \\ 1 \\ b \\ 0 \\ b \end{array}$
- **Port 2:** $\begin{array}{c} 0 \\ b \\ a \\ 1 \\ c \\ 1 \\ b \\ 0 \\ b \end{array}$
- **PIs:** $<1, \{1,2\}>, <2, \{2\}>, <2, \{1\}>, <1, \{1\}>$

**Tester 1 – Port 1**
**Tester 2 – port 2**
**The initial state is A**
Protocol Performance Evaluation

- Objectives of Performance Evaluation Study:
  - detect unanticipated phenomena of the system in order to avoid malfunction.
  - give precise knowledge about the resource distribution throughout the system.
  - allow for identifying the bottlenecks.
  - find an optimal resource configuration.
  - maximizing the efficiency.
  - avoiding the purchase of unnecessary equipment.
Performance Testing

• What to measure?
  – Message response time:
    • terminal response time.
    • overall response time.
  – Throughput:
    • sum of all packets received at their destinations successfully (per unit time).
  – Reliability:
    • percentage of time a user can communicate with required points in the network.
  – Cost
  – Other Parameters:
    • network sensitivity to peak load traffic,
    • packet route blocking probability and
    • transmission error rates, availability of resources,
    • expandability and ease of use
Performance Test Methods

• analytic modeling
  – construction of a model.
  – a mathematical construct.

• If the mathematical method model of a protocol is tested, following results are obtained to test the performance of the protocol.

• Bounded calculation: These are simple back of the envelope calculations that are done to quickly assess the maximum throughput of a communication protocol.

• For example consider the slotted csma/cd, we get

  Maximum throughput as $\frac{1}{1+3.31\beta}$ when $g(n)=0.77$. 
- **simulation modeling**
  
  - a specialized computer program.

- **performance measurement (benchmarking)**
  
  - relies on direct observation of the system of interest, or a similar system.
Protocol performance testing by analytical methods

- Queuing network models
  - represents a set of services that process data (also known as jobs, transactions, or arrivals).
  - predict the latency, throughput and utilization of a system and its components.
  - predict communications system availability given component failure rates and repair (recovery) rates.

- To compute performance of Unslotted CSMA/CD protocol, channel can be modeled as
  - single queue with one buffer and can service only one buffer
  - The data is dropped if two workstations send the data simultaneously (collision).
  - The workstations randomly backoff and transmit again in the next slots.
  - channel can be modeled as a single queue with multiple Markov chains can also be used to model communication systems as a finite set of states with known rates of transition between states.
Real Time Tests

- Motivations for measuring system performance:
  - Marketing
  - Product Validation
  - Model Validation
  - Support Performance Tuning
  - Data Collection
  - Trace Collection
  - Selecting a Workload
Real Time Tests

• Implementing workloads
  – three basic elements:
    • the system under test (SUT),
    • load generators (driver) that apply a workload to the SUT, and
    • instrumentation to collect performance data.

• Load generation
  – special purpose hardware,
  – software running on separate system,
  – process running on the SUT.

• Performance instrumentation
  – measure latency and throughput,
  – identifying bottlenecks.
Simulation

• models a real-world system as a computer program.

• Simulation allows a protocol to be modeled at any level of detail:
  – from a direct translation of a queuing network model to capturing every aspect of the protocol's behavior.

• Simulation also supports the collection of any performance metric that can be defined and programmed.

• Architecture of simulations
  – discrete event simulations
    • performance analysis
    • the representation of time is quantified, and
    • the system state only changes when an event occurs.
  – continuous simulation
    • models time as a continuous progression.

• Basic techniques to generate workloads for simulation:
  – Stochastic, Trace-driven, or Execution-based.
Interoperability Testing

- Interoperability testing measures if two or more protocols correctly implement the technical specifications necessary to ensure successful integration supporting particular communication protocols.
- The verification of interoperability is the process of demonstrating a successful communication between tested protocols.

Two views:
• considers the interoperability as an absolute concept.
  Applying measures of communications system's interoperability with different systems and document them separately
Interoperability Testing

a) Complementation

b) Correspondence

c) Followness

An example of Conformance and Interoperability test suites correlation.
Interoperability Testing

Stimulus (Request)
   Response (Indication)
Stimulus (Response)
   Response (Confirmation)

(a) Confirmed Services

(b) Unconfirmed Services

The required stimulus-response pairs in (a) confirmed (b) unconfirmed services
Interoperability testing of connection establishment protocol

Connection time sequence diagrams
Scalability testing of BGP

The following are the test procedures used in scalability testing of BGP.

Test case 1: Initially a baseline set of tests was performed which determined the number of routes which could be held in a VRF (Virtual routing and forwarding) table relative to the number of VRFs that were configured on the router.

Test case 2: Once the baseline set of tests has been performed the configuration on the router was altered so that each VRF had a firewall filter and a counter on its outgoing interface. This subset of tests was performed with 5, 10 and 15 filter statements and counters per VRF.

Test case 3: The third set of tests involved altering the configuration again by adding a policer to each VRF in addition to the filter and counter added in test case 2. Once this was established a similar subset of tests to test case 2 was performed.
Scalability Testing of BGP

BGP Layer 3 MPLS VPN network
Network Scenario Test

Simulated CE Routers
E-BGP Routes from RT
OC-48 Link

Simulated PE
Simulated Provider Network
Simulated PE

Router Tester
M20 (Juniper)
OC-48 Link

Router Tester
M40 (Juniper)
OC-48 Link

Traffic Flow

Network Scenario Testing configuration.

RouterTester
simulated OSPF grid
(50 routers)
SDL Based Performance Testing of TCP

• Procedures for performance testing using SDL:

  - Create FSMs of processes of TCP.
  - Draw SDL specifications of the FSMs.
  - Apply the inputs to the specifications and run the simulation.
  - Create a message sequence chart as well as record the performance by using some performance variables.
SDL Based Performance Testing of TCP

continues

MSC obtained for TCP performance testing
SDL Based Performance Testing of TCP

• Results:
  – By applying the startsim signal the simulation was run for max of 200 pkts transmission and an MSC is created (figure is a part of complete MSC).
  – sender exponentially increases the window size after receiving ACK signal.
  – reduces the window size to 1 whenever a timeout of the packet occurs.
  – The simulation can be run for any number of packets by changing the counter value in the Transmitter process.
  – The throughput obtained for the given specs (200 pkts, timeoutperiod= 400 msec, max window size = 8) is 50% for random loss created in the channel.
  – Average delays for a packet observed are 200 msec.
  – Varying timer values: higher delays and less throughput.
  – Varying window sizes: Throughput and delays were reduced with random loss model since the model considered creates losses of upto 40%.
SDL Based Performance Testing of OSPF

- The performance data quantifies the amount of traffic an OSPF IP network generates during the time intervals when routing tables are refreshed.

- A simple case to illustrate 'flooding' concept
  - A topology of 11 nodes.
  - Network is simultaneously flooded at 30 min intervals.
  - Number of LSA packets generation time and number of LSAs are observed for different cases:
    - routers started simultaneously and each router has random LSAs to be sent at random time,
    - routers starting randomly and each router has random LSAs to be sent at random time, and
    - routers started simultaneously which have LSAs to be sent at the same time.
  - Observations: total number of LSAs sent for case 1 and 2 are distributed in time and reduce instant communication overheads whereas for case 3 number of LSAs to be sent are concentrated at a particular small time interval increasing the instant communication overheads. The number of LSAs generated are about 200.
SDL based interoperability testing of CSMA/CD and CSMA/CA protocol using bridge

- The procedure for interoperability testing by using SDL is as follows.
  - Create FSMs for the protocols and their interoperations using a bridge.
  - Create SDL diagrams of the protocols and the bridge operations.
  - Run the system by giving the inputs (test sequence) from the environment such as source, destination and number of packets.
  - Create the MSCs for the different kinds of inputs.
  - Observe from MSCs interworking of CSMA/CA and CSMA/CD using a bridge.
MSC Interoperability test 2

user4
outpac
req_chan_sta
res
packet21
fpacket1
packet
end_packet

req_chan_state
res
packet
packet_bridge

packet

packet

packet
Results

1) MSC for the movement of 3 packets from node 1 (on CSMA/CD) to node 3 (on CSMA/CA). The bridge checks the destination of the packet, if it is on the same link, then it will not forward the packet.
   
   • In case destination is on different link, the bridge buffers the packet until CSMA/CA channel is free.
   
   • Once the channel is free, packets are sent one by one to the receiver.

2) MSC for the movement of 3 packets from node 4 (CSMA/CA) to node 2 (on CSMA/CD).
   
   • the bridge is forwarding packets reliably to the node 2.

3) MSC for the movement of packets in both directions simultaneously (from node 1 to 3, and node 4 to 2).
   
   • bridge is able to handle the data transfer reliably for both sides.