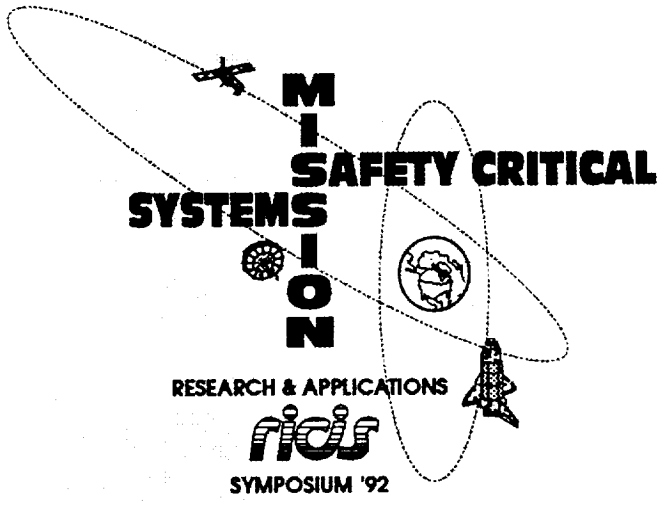
N95-14163

27237 22 p.

19951077749



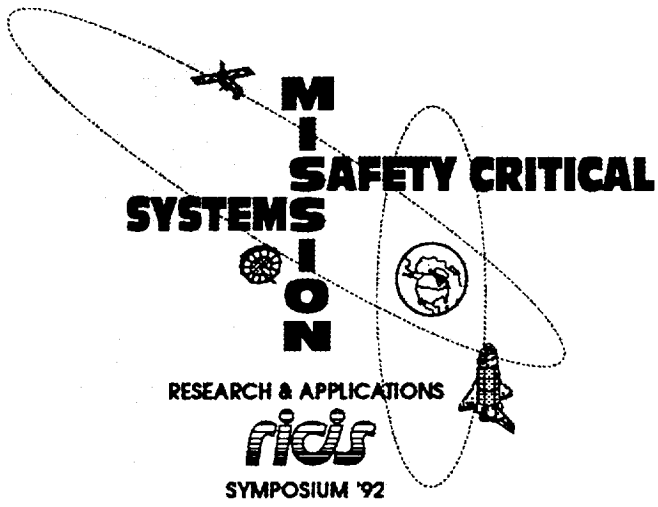
CAPTIONALS: A COMPUTER AIDED TESTING ENVIRONMENT FOR THE VERIFICATION AND VALIDATION OF COMMUNICATION PROTOCOLS

C. Feng, X. Sun, Y.N. Shen and F. Lombardi

ABSTRACT

This talk presents novel issues involved in the verification and validation of protocols for distributed computer and communication systems using a computer aided testing approach (CAT). Verification is the process which substantiates the accuracy of the specifications of a protocol; validation is the process in which the validity of the specifications to meet the desired objectives (or requirements) is confirmed. They make up the so-called process of conformance testing. Protocol implementations which pass conformance testing, are then checked whether they can operate together. This scenario is referred to as interoperability testing.

A new comprehensive approach to protocol testing is presented. This approach addresses new fundamental issues for testing protocols: (1) modeling for inter-layer representation for compatibility between conformance and interoperability testing. (2) computational improvement to current testing methods by using the proposed model inclusive of formulation of new qualitative and quantitative measures (such as detectability) and time-dependent behavior. (3) analysis and evaluation of protocol behavior for interactive testing without an extensive use of simulation. These problems careful require definition and are analyzed using the proposed CAT approach: (1) modeling of protocol activities at different levels of abstraction (inter-layer) to facilitate a cohesive approach to both conformance and interoperability testing through the use of a new analytical model (based on cellular automata); (2) design of a set of tools for interactive



testing by analytical techniques with partial reliance on simulation. (3) evaluation of new qualitative and quantitative measures for protocol testing through the development and use of appropriate interactive analytical tools as well as a testbed evaluation. The applicability of the proposed approach to real-life protocols (such as the abort sequence for the Space Shuttle) will be presented.

FABRIZIO LOMBARDI

BIOGRAPHY

Fabrizio Lombardi was born in Formia (Italy). He graduated in 1977 from the University of Essex (UK) with a B.Sc. (Hons.) in Electric Engineering. In 1977 he joined the Microwave Research Unit at University College London, where he received the Master in Microwaves and Modern Optics (1978), the Diploma in Microwave Engineering (1978) and the Ph.D. (1982). He is currently an Associate Professor in the Department of Computer Science at Texas A & M University. He was the recipient of the 1985/86 Research Initiation Award from the IEEE/Engineering Foundation. In 1988, he was awarded the Visiting Fellowship at the British Columbia Advanced System Institute, University of Victoria. Dr. Lombardi was also a co-director of the NATO Advanced Study Institute on Testing and Diagnosis of VLSI and ULSI, June 1987, Como (Italy). His research interests are verification and validation of protocols, fault tolerant computing, VLSI testing, and real-time systems. Dr. Lombardi is a Distinguished Visitor of the IEEE Computer Society and a Research Fellow of the Texas Engineering Experiment Station. Dr. Lombardi is the Program Chair of the 1992 IEEE International Workshop on

CAPTIONALS: A Computer Aided Testing Environment for the Verification and Validation of Communication Protocols

By

C. Feng, X. Sun, Y.N. Shen

and

F. Lombardi

Texas A&M University
Department of Computer Science
College Station

Tel: (409)845-5464

Environment

Computer

Aided

Protocol

Testing by

Input

Output Behavior and By

Numeric

And

Logic

Specifications

- **Deterministic Behavior**

Protocols

- Complex entities/combination of hardware and software with no perceived separation of the two.
- Black box characterization
- Operation specified by international standards
- Interoperability among multiple protocol entities.

Important Issues

- Incompleteness in specifications
- Ambiguities in behavior due to inherent complexity
- Limited controllability and observability in the sequencing of the operation of the protocol entity
- Control the state explosion phenomena for representing a protocol (or combination of protocols)
- Ease of use for interactive operation

Previous Environments

- Postman by AT&T Bell Labs :
  (Graph Based Approach)
- Estelle:
  (Semantic Based Approach)
- OSI/ISO convention:
  (SDL Approach)
- Main disadvantages: restricted applicability; no generation of test sequence, only proof of existence of particular properties of a protocol; complex operation, not suitable to non-expert user; heavy computational requirements.

Major Objectives

- Verify and validate hardware/software computing/communication systems
- Generate sequence for conformance testing to specifications
- Specification identification using logic and numeric semantics
- Identify and eliminate ambiguities in specifications and guarantee safety
- Evaluate fault coverage and identify components which yield loss of coverage
- Testbed evaluation

Computer Aided Testing

- Set of (semi-automatic) tools provided to a non-initiated user
- On-Line (interactive) evaluation of different testing strategies
- Experimental validation by testbed (yet to be implemented using NSF funded SI grant)
- Conformance to international standards (for OSI through ESTELLE)

Organization

- **Modeling**

  Protocol specified in terms of basic steps (atomic actions).

  Atomic actions performed by abstract machines.

  Each abstract machine defined with respect to a single attribute.

  Machines combined through a dependability net to represent overall behavior through a multi-level representation. (hierarchical representation)

  Advantages: it separates requirements from specifications, reduces computational overhead due to simulation, allows consistency checking.

Organization (Continued)

- **Evaluation**

  Generate using analytical techniques (on-line) figures of merit for test sequence evaluation and generation.

  Performed using the trace of an abstract machine, or the dependability net of the automata.

Organization (Continued)

- **Tools**

  Provide data model to capture protocol entities and their relationships (identification of operational features, composition and instantiation to define higher level of abstraction, generalization to support certification procedures).

  Translate attributes as evolution of structures using dynamic algebra semantics.

Numeric Specifications

- Applicable to dependency due to sensor data and timing specifications.
- Discrete (bounded) range validated only at critical values.
- Reachability analysis for establishing critical paths in protocol execution.
- Reference to a master control must be provided to handle consistency.

Logic Specifications

- Interdependences in stimula (inputs) on the behavior of the protocol.
- Sequentiality of protocol execution broken at single specification to allow manipulation of logic expressions (as enabler) through minimization.
- Decomposition of the protocol by isolating logic (correlated) behavior.
- Minimization using traditional VLSI-CAD tools (e.g., Quine-McCluskey)

Example: Conditional Time-out

Assume a periodic task over 100 cycles (clock incremented on a single cycle basis).

Specifications:

Keep the green light on and the red light off provided either the green switch is not pressed during the first 50 cycles or the yellow switch is not pressed during the second 50 cycles else, turn off the green light and on the red light if a switch is pressed and ring a bell.

To turn off the red light and turn on the green press the yellow switch and ring a bell during the first 50 cycles.

Reset capability is provided.

Example: Conditional Time-out

Protocol Representation

Two States:

S_1 = Green light on, red light off

S_2 = Green light off, red light on

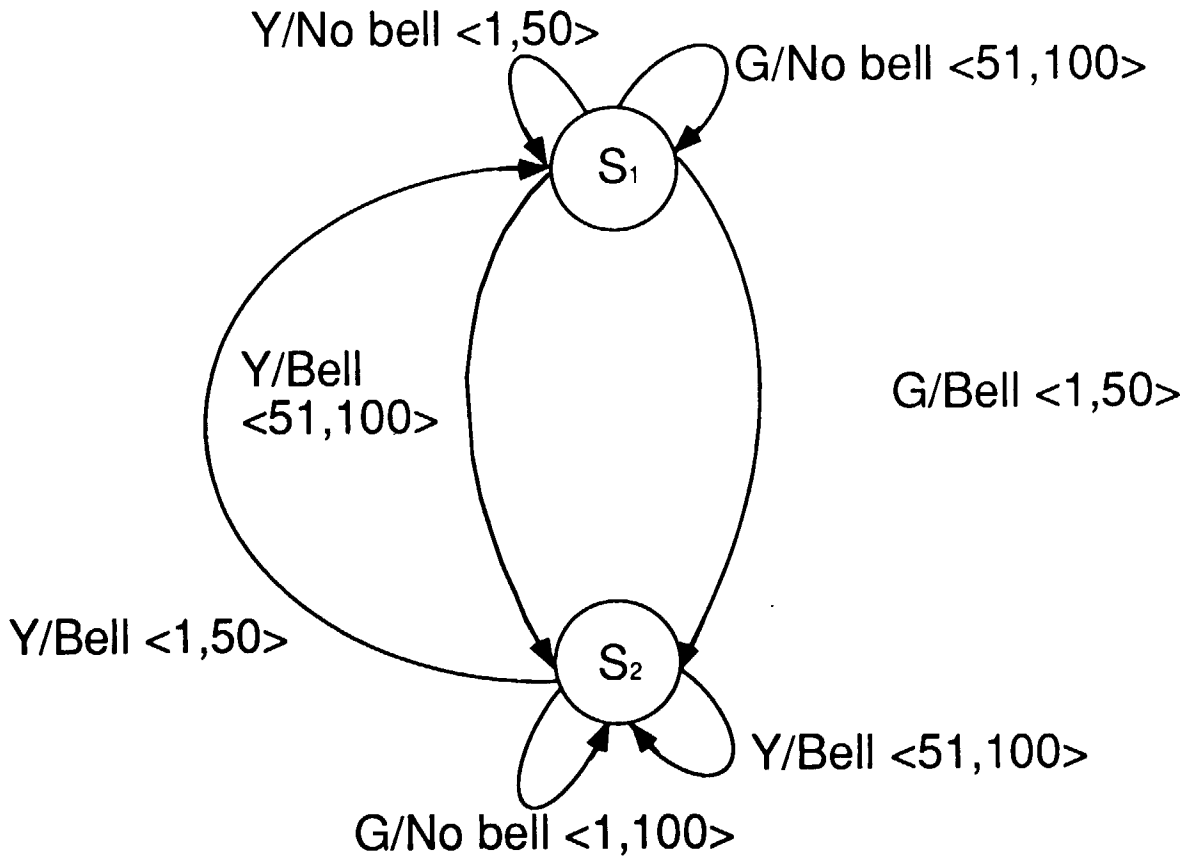
Output: Bell (or no bell)

Input: Yellow switch (Y),
Green switch (G)

Label: Input/Output

Clock: counter from 1 to 100 (and back)

Graph Representation



T = <1,100>

Reset edges not shown

Test Approach

- Touring: tour every edge of the representation at least once.
- Suite: generate a test subsequence for each type of fault which is allowed in fault model.
- Traverse an edge at lower and upper timing specifications using unique input/output qualifiers for the transitions (actions).
- Minimality achieved using a minimum spanning tree of the underlying time graph (by splitting each edge into multiple parallel edges to satisfy timing constraints).

Features

- Functional fault model (faults in logic and timing specifications)
- Test sequence generation as functions of inputs and time interval
- Ambiguity identification: press both switches (safe behavior due to single transition in protocol representation)
- Critical timing: 50th and 51st clock cycles
- Safety: both lights on (or off)

Evaluation

- Check every specification on both timing bounds.
- For interoperability: assume two equal protocol entities (X, Y), X as initializing the process.
- Bell output of first protocol (X) corresponds to green switch on of the other protocol (Y) and vice versa (yellow switch still a primary input).
- Assume only one protocol implementation to be faulty; then full controllability is lost (lost of timing specifications).
- Observability still preserved due to disjoint nature of the two switches.

Example 2:
Shuttle Abort Sequence Protocol

- Abort QMS/RCS Interconnect (4.18a)
- Presence of both logic and timing dependencies
- 16 steps
- Very sparse protocol (i.e., specifications are based on a large number of inputs and outputs), thus very good observability
- Incomplete specifications
- Generated test sequence and validated under different fault conditions.
- Verified that the sequence in the specification yields a termination of the procedure.

Results

- Validated and verified the following protocols:

  X.25

  SNA

  ABP

  LLC 802 Token Ring

- Average saving in the number of tests is almost 20% (compared with Postman)