
1st International Workshop on Reliable Cyber Physical Systems
Irvine, October 8th, 2012

Grit Denker, Daniel Elenius, Carolyn Talcott
SRI International
Gene Zhijing Qin, Nalini Venkatasubramanian
University of California Irvine
Overview

• Objective of Formal-Method Based Analysis

• Analyzed Network Types
  – Multinetworks
  – Time-Triggered Ethernet

• Summary
Objective: Integrated Fault and Performance Analysis

Closed Loop Control Systems
  e.g., x-by-wire systems, Engine Controls, Power Systems, Sensor-controller-actuator systems

Real-time Audio Systems
  e.g., Voice over IP (ITU-T), G.729 etc

Real-time Video Systems
  e.g., Compressed/Uncompressed video

COTS (BE), IP Data
  e.g., Web applications, FTP etc

Integrated System Network

Integrating CPS subsystems with varying criticality levels and varying fault-tolerance and performance requirements over a common network

Design and Verification Objectives:
• Performance (latency, jitter, bandwidth) requirements of each application are met
• Fault tolerance requirements for each subsystem are met in presence of failure of network/host components
• Expose emergent behavior that may invalidate system assumptions and requirements

"Co-optimization" with verification proofs over general space of architecture is intractable

Our approach: Analysis for selected points in architecture space or selected network configurations to gain insights into tradeoffs
Multinetworks

**Objective:**
Use up-to-date network state info to model and analyze network configurations

**Approach:**
Perform formal “What If?” analysis on network model

**Benefits:**
Insights from analysis provides guidance for network administrator

Multinetwork Information Architecture (MINA)
Centralized DB holds the network state information collected from devices and links
(Luca Iannario, Gene Qin et al, UC Irvine)
What If Analysis

• Which flow will be affected if node is down? (Node Criticality Index)
• What happens if load of selected flows is changed?
• What happens if link quality changes (e.g., congestion, interference)?

Node Criticality Index
For each network node
• Assume node goes down
• Reroute all flows
• Analyze QoS parameters of affected flows

Use node criticality index to decide use of network resources in reconfigurations so that likelihood of application requirement satisfaction will increase

Compare with conventional approaches
Conventional Approaches

Node Centrality
• Node degree
• Closeness-based (find center)
• Betweenness (check shortest path/flow load passed through)

Hypothesis: Centrality-Based Techniques not suitable for dynamic and heterogenous networks

Backup and Reconfiguration Planning
• Work-load based backup
• Load balanced-based-based reconfiguration

Hypothesis: These approaches do not exploit impacts across redistributed flows

Network Planning and Simulator-Based Approaches

Hypothesis: More time consuming than high-level formal model
Experiment – Backup Nodes: Setup

- Reroute affected nodes using Dijkstra algorithm
- Compute impact of failing node on delay of flows
- Compare critical node index with centrality index

- Constant bit rate applications to describe flows
- Analyze end-to-end QoS parameter for each flow, assuming one node down
- Do analysis for every node

<table>
<thead>
<tr>
<th>ID</th>
<th>Route</th>
<th>Type</th>
<th>TP</th>
<th>Length</th>
<th>Num</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1,2,25</td>
<td>1</td>
<td>0.8</td>
<td>0.0016</td>
<td>500</td>
</tr>
<tr>
<td>2</td>
<td>1,5,26</td>
<td>1</td>
<td>0.8</td>
<td>0.002</td>
<td>400</td>
</tr>
<tr>
<td>3</td>
<td>19,7,3,11</td>
<td>1</td>
<td>0.8</td>
<td>0.0016</td>
<td>500</td>
</tr>
<tr>
<td>4</td>
<td>21,3,4,13</td>
<td>1</td>
<td>0.48</td>
<td>0.0016</td>
<td>300</td>
</tr>
<tr>
<td>5</td>
<td>1,6,10</td>
<td>2</td>
<td>0.48</td>
<td>0.0016</td>
<td>300</td>
</tr>
<tr>
<td>6</td>
<td>24,21,11,17,14</td>
<td>2</td>
<td>0.4</td>
<td>0.002</td>
<td>200</td>
</tr>
<tr>
<td>7</td>
<td>23,26,5,2,6,10</td>
<td>2</td>
<td>0.4</td>
<td>0.002</td>
<td>200</td>
</tr>
<tr>
<td>8</td>
<td>26,5,7,12,16</td>
<td>2</td>
<td>0.24</td>
<td>0.0008</td>
<td>300</td>
</tr>
</tbody>
</table>
Assume limited backup resources
Choose backup node using criticality or centrality
-> backed up nodes won’t fail
Let other nodes (not src/dest of flows) fail for 300ms, and determine overall impact on delay of all flows
Experiment II – NW Reconfiguration

- Node 2 is down
- Nodes 25 & 26 can re-associate to either node 5 or 6
- Simulate four possible new configurations to rank them
- Compare rankings with Qualnet ranking

<table>
<thead>
<tr>
<th>NW Config</th>
<th>Avg Delay Increase</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>NW1</td>
<td>54.0502%</td>
<td>3</td>
</tr>
<tr>
<td>NW2</td>
<td>27.6939%</td>
<td>1</td>
</tr>
<tr>
<td>NW3</td>
<td>65.6767%</td>
<td>4</td>
</tr>
<tr>
<td>NW4</td>
<td>28.5294%</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NW Config</th>
<th>Load Difference</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>NW1</td>
<td>1.22</td>
<td>4</td>
</tr>
<tr>
<td>NW2</td>
<td>1.02</td>
<td>3</td>
</tr>
<tr>
<td>NW3</td>
<td>0.58</td>
<td>2</td>
</tr>
<tr>
<td>NW4</td>
<td>0.38</td>
<td>1</td>
</tr>
</tbody>
</table>

Qualnet vs Formal Method
- More complex configuration, less reuse
- Redundant operations: e.g. setup subnet, configure ip, etc.
- Slower at runtime
Time Triggered Ethernet (TTEthernet)

- Used in safety-critical, real-time CPS such as aircrafts and automobiles
- Has three message types: Time-triggered, rate-constrained and best-effort
- Three types of components: Host, End System (ES), Switch (SW)
  - Each has Tx (transmit) and Rx (Receive) functionality
  - Each as standard (SI) and high (HI) integrity
    - HI with duplicated processing between Rx and Tx
- Achieves fault tolerance via high integrity and path and system redundancy: But how much is sufficient and necessary?

If path redundancy, every dataflow is broken into 2 or 3 channels with disjoint physical links
Network Architecture Tradeoff Analysis Tool Chain

- **System and Traffic Inputs**
  - Applications
  - Traffic Profiles
  - System Level Requirements

- **Network Architectural Choices**
  - Topology
  - Integrity
  - Availability

- **Architecture Specific Choices**
  - Synchronization
  - Mixed Mode Traffic Classes

**Performance Fault Free Analysis Tools**
- RC: Check feasibility
- TT: Check schedulability

**Probabilistic Fault Analysis Tool**
- Check Fault Tolerance Properties

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Fault Probabilities</th>
<th>TT Schedule</th>
<th>RC Latency</th>
</tr>
</thead>
<tbody>
<tr>
<td>HI, TT only, 75% load on nw</td>
<td>(1e-4,0,...,1e-18)</td>
<td>Yes</td>
<td>n/a</td>
</tr>
<tr>
<td>HI, TT only, 90% load on nw</td>
<td>(1e-4,0,...,1e-18)</td>
<td>No</td>
<td>n/a</td>
</tr>
<tr>
<td>HI/SI, TT/RC, 90% load on nw</td>
<td>(1e-4,0,...,1e-18)</td>
<td>Yes</td>
<td>VL1&lt;x, VL2&lt;y</td>
</tr>
<tr>
<td>SI, TT only, 50% load on nw</td>
<td>(1e-2,1e-3, ...)</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Brake-by-Wire Fault Analysis and Performance Checks

Analyzed Faults: silent, omission, commission, invalid fields, untimely

Safety Requirements
- If the brake pedal is engaged, brake at each wheel
- If wheel brake is engaged, illuminate brake lights
- If wheel brake is engaged, close throttle at motor
- If brake light doesn't work, show warning in driver display
BBW Schedulability, Utility and Fault Results

TT Schedule with 35 additional Frames

Approximate the end-to-end latency for given topology

<table>
<thead>
<tr>
<th>Arch</th>
<th>VL ID</th>
<th>Fault Probabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>All HI</td>
<td>1</td>
<td>OM: 1.3e-4, COM: 0, VSA: 0, VDA: 0, VSN: 0, VLEN: 0, VDATA: 1.0e-8, VFCS: 1.0E-10, TE: 0, TL: 0</td>
</tr>
<tr>
<td>All SI</td>
<td>1</td>
<td>OM: 1.3e-4, COM: 1.0e-5, VDA: 1.0e-5, VSN: 0, VLEN1.0e-5, VDATA: 2.001E-5, VFCS 1.0e-5, TE: 1.0e-5, TL: 1.0e-5</td>
</tr>
</tbody>
</table>
Summary

• Formal what-if analysis
  • Is suitable for analyzing faults in heterogeneous networks
  • Can account for impact across flows
• Co-optimization of fault tolerance and performance with verification proofs over general space of architecture is intractable
• Tradeoff analysis for selected points in design space is feasible
• Integration of latency, utilization, buffer size and fault tolerance analysis is scalable to vehicle-sized network architectures

Support architecture design or network configuration choices through formal methods-based analysis