Layered Dependability Modeling of an Air Traffic Control System

Olivia Das and C. Murray Woodside
Department of Systems and Computer Engineering
Carleton University, Ottawa, Canada
odas@sce.carleton.ca, cmw@sce.carleton.ca
Overview

• dependability of complex systems

• dependability for systems with layered software architecture

• effect on coverage due to management subsystem failures

• performability measures
Layered Application Model

Tasks, Interactions and Dependencies, and Processors

Controller (up to four controllers)

Display Management

Flight Plan Management

Flight Plan Database

Conflict Resolution

Trajectory Management

Surveillance Processing

Synchronous service request

Asynchronous service request

(upper four controllers)

(two radars)
Replication Mechanisms

Primary-standby, load-balancing, active, primary-standby-active

\[ N_{\text{UserA}} = 50 \]
\[ N_{\text{UserB}} = 100 \]
Example Configuration (1)

proc3 fails and causes Server1 failure...Server2 used instead
Example Configuration (2)

proc1 fails and puts AppA out.. Group UserA fails..
Here, failure cannot be compensated by standby servers
Centralized Fault Management Model

Components
- Application Tasks
- Mgmt. cmpts.

Connectors
- Alive-Watch
- Notify
Perfect detection and reconfiguration

proc3 fails and causes Server1 failure...

*Full coverage: Server2 used instead*
Partial coverage for centralized mgmt.

proc3 fails and causes Server1 failure...

Partial coverage: Manager failed, so system failed
Analysis - currently

Level 1

Determine Distinct Operational Configurations $C_i$

Compute Probability, $Prob(C_i)$, of each Operational Configuration

Level 2

Compute Reward, $R(C_i)$ of each Operational Configuration using Layered Queueing Models

Compute Mean Reward = $\sum_{i} R(C_i) \cdot Prob(C_i)$
Probabilities of Operational Configurations

Layered Application Model

Fault Propagation Graph (AND-OR)

Knowledge Propagation Graph (directed)

Non-coherent fault tree
Layered Model of ATC En Route System
Fault Mgmt. Model of ATC En Route System

Console Processor

Central Processor

Radar Processor

Name Server Processor

Monitor and Control subsystem (Three active replicas)
Results

Number of components (tasks and processors): 51
Number of connectors in fault management model: 118
Failure probability of all processors: 0.05
Failure probability of all tasks (including management tasks): 0.1

Total number of nodes in the graph that combines information from both the fault propagation graph and the Knowledge Propagation graph: 715
Number of operational configurations: 14
Time to generate and compute probabilities of configurations: 277 secs
Probability of system being in working state: 0.33
Average throughput for Controller task: 0.067 requests/sec

If failure probability of management tasks decreased to 0.05, then
Probability of system being in working state: 0.45 and average throughput for Controller task increases to 0.093 requests/sec.
Conclusions

• Dependability evaluation for layered software architectures

• Scalable technique

  • separation of performance analysis from failure-repair
  • much smaller set of configurations because of layered architecture than of failure states

• Operational configurations takes into account:

  • layered dependencies
  • "Knowledge failure" effects that depends on the status of the Management system which limits the reconfiguration capability

• Explosion of configuration is a limitation