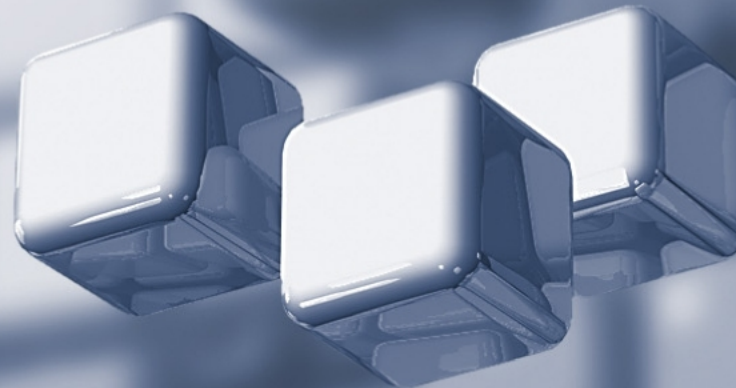


The Automated Generation of Test Cases using an Extended Domain Based Reliability Model



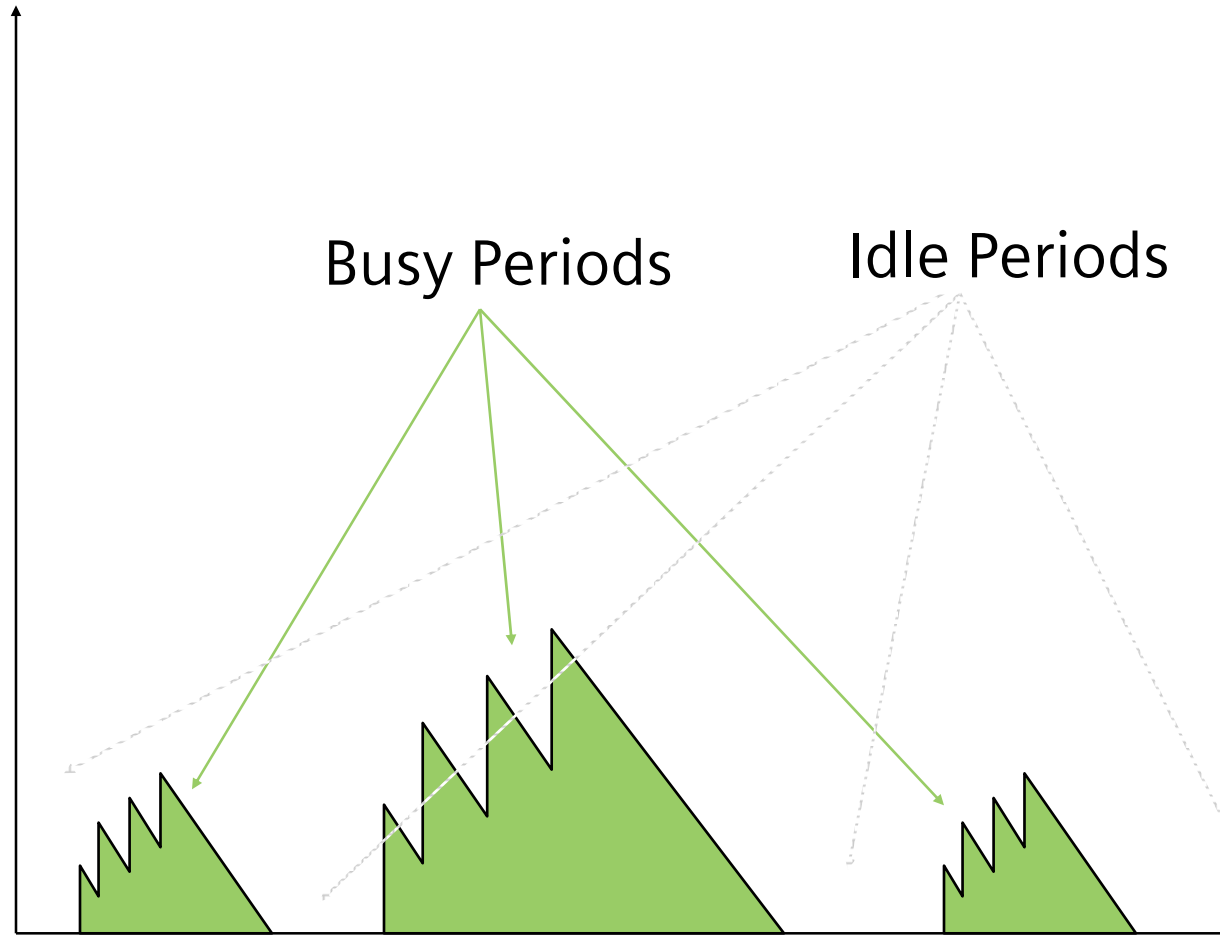
Performance/Reliability Testing Objectives

- Performance/reliability estimation in system test
- Uncover resource contention issues (e.g. kernel usage, deadlocks, memory leaks)
- Automated test case generation and execution



Unfinished Work in Statistically Based Testing

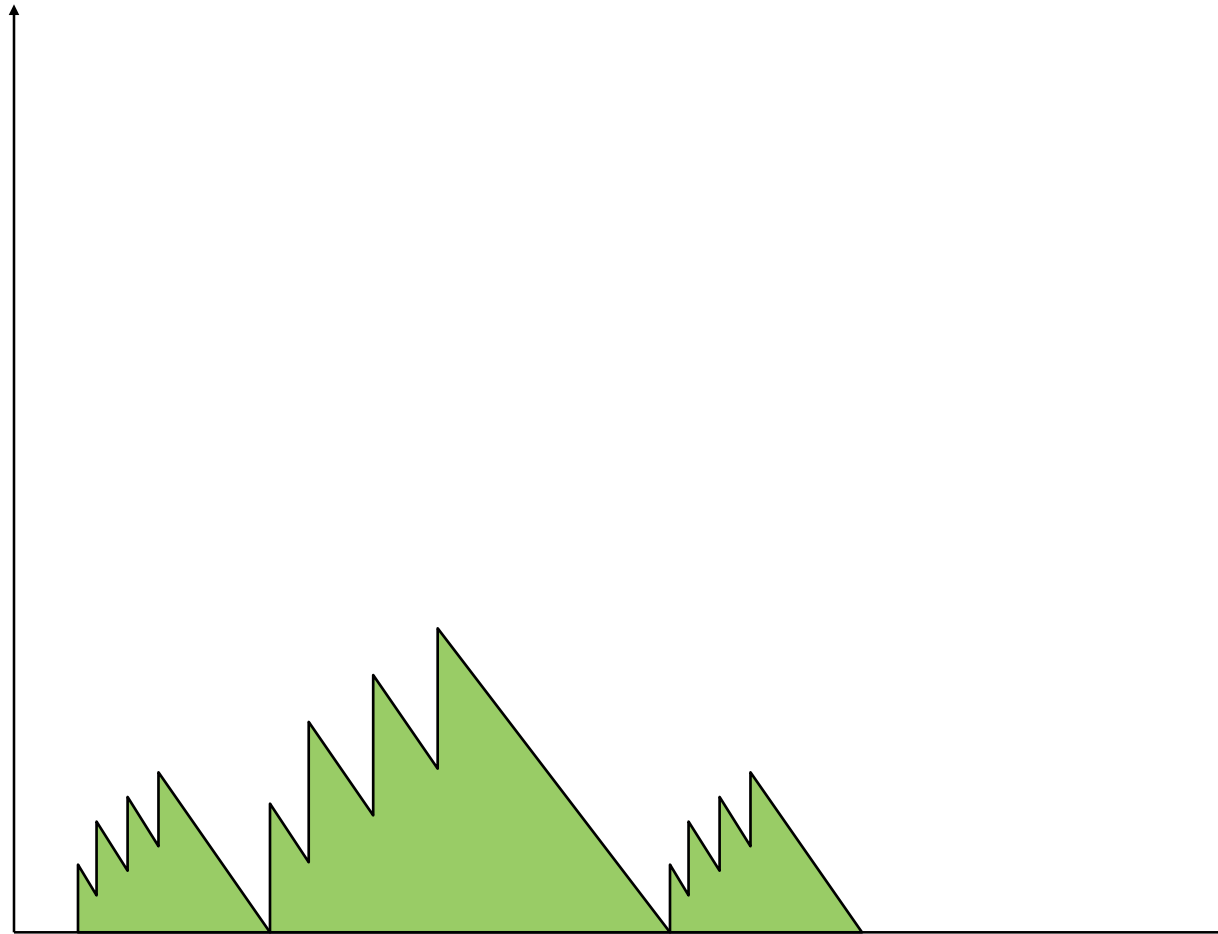
Unfinished Work



Unfinished Work and the Busy Period

Markov State Testing Approach

Unfinished Work



Unfinished Work and the Busy Period

Markov State Testing Approach

- Modeling assumptions:
 - Poisson arrivals and exponentially distributed call holding times
 - Average arrival rate and holding time can be obtained
 - System operates at low to medium utilization <50%
 - Each call type is ~ as an independent m-server loss queuing system (M/M/m/m)

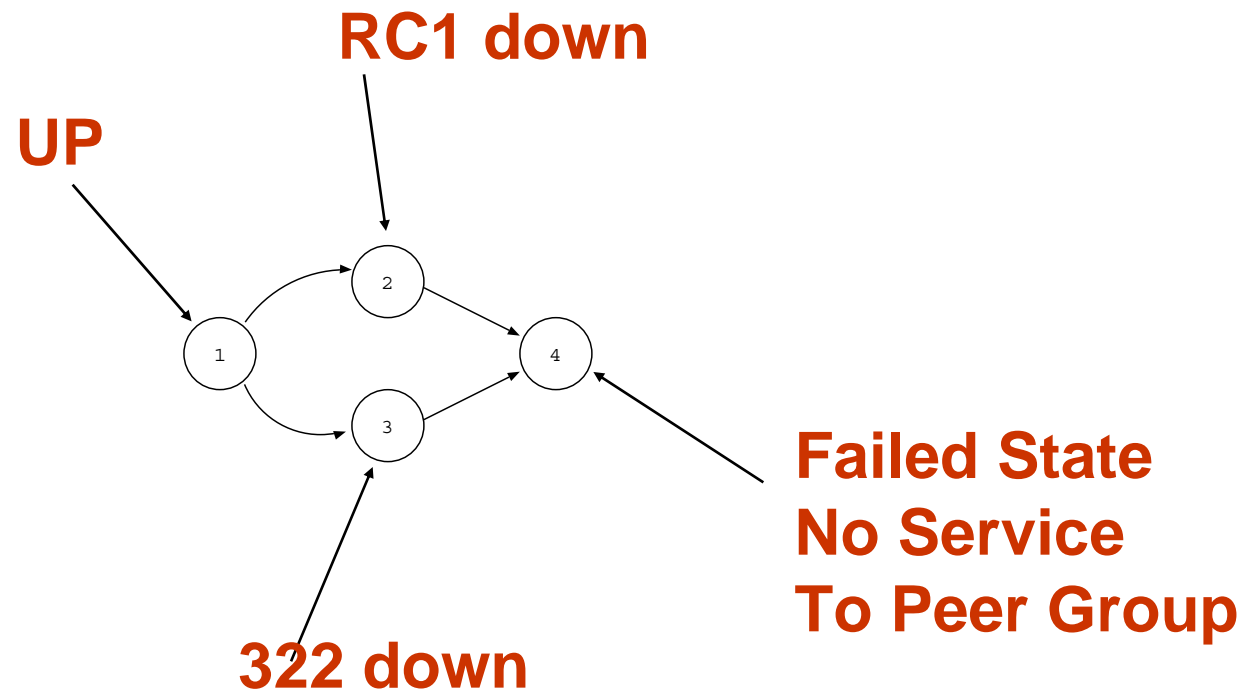
Markov Testing Approach (TSE 95)

- State definition at the right level of abstraction (e.g. number of active calls of each type)
- Joint distribution \sim by the product of the independent probability distributions
- Take advantage of the shape of the probability distribution to generate all the states with probability greater than an empirically configured epsilon
- Starting from the idle state, drive the system deterministically into each state under test

- Issue: Does not account for reliability decay as a function of time

- *The Automatic Generation of Load Test Suites and the Assessment of the Resulting Software, IEEE TSE, Sep 95*

Failure Based Markov Chain - Showing two paths to a failed state



Reliability estimation accounting for Mission Time and Service Impact of Faults

$$R(t) = 1 - \sum_i \left(P_t(i) * \frac{N_{fail}(i)}{N_{total}} \right)$$

-Reliability Estimation for Large Distributed Software Systems, CASCON 2008

Extended Domain Based Reliability Model

- Define two Markov Chains:
 1. Resource usage Markov chain (as introduced in TSE Paper)
 2. Failure based Markov chain (as introduced in Cascon Paper)
- Use decomposition/aggregation to obtain the Reliability estimate

Extended Domain Based Reliability Estimation Process



1. Define failure-based system state $F1$ and execute the algorithm to generate a test suite $TF1$,
2. Use a Markov chain solver to obtain the transient solution of the Markov chain, for the specified system execution time,
3. Define resource usage-based system state $R1$ and execute the algorithm to generate a test suite $TR1$,
4. For each state $R1S$ in the test suite $TR1$, evaluate pass/fail condition for the state $R1S$ for each state in test suite $TF1$
5. To obtain the reliability assessment for the system, we compute the weighted sum of the state probabilities that are failure-free. This step consists of the application of Bayes Theorem.

Extended Domain Based Reliability Estimation Process



In the system execution time-based reliability modeling approach, we let the set of failure-based states that have been tested be denoted by $1, 2, \dots, y$.

We let the set of resource usage-based states that have been tested be denoted by $1, 2, \dots, z$. We let s_{ij} denote the intended state corresponding to the i th resource usage-based test case, when resource failure j is active.

$$R[p, S, t](P) = 1 - \sum_{(i=1, z)} \sum_{(j=1, y)} p(i|j) p t(j) \alpha(i, j)$$

$$R[p, S, t](P) = 1 - \sum_{(i=1, z)} p(i) \sum_{(j=1, y)} p t(j) \alpha(i, j)$$

where,

$$\alpha(i, j) = 0 \text{ if } P(i, j) = s_{ij}, 1 \text{ otherwise.}$$

Domain of Applicability

- Reliability Estimation as a function of mission time for systems for which performance is a function of the number of available resources:
 - Telecommunication switches (pots)
 - Servers that are required to support large volumes (e-mail servers)
 - Class C reliability systems with hardware replication (multiple instances of hardware and software components)

Example of Resource Allocation and Deallocation Data



Resource Type	Avg Arr Rate (calls/min)	Avg Hldg Time (Minutes)
OP1	3.3	3.0
OP2	0.823	3.0
OP3	0.588	2.0
OP4	0.297	1.0
OP5	0.588	5.0

© Siemens Corporate Technology (CT) and Siemens Corporate Research (SCR), 2008 all rights reserved

Example of Resource Failure and Repair Data

Resource Type	Failures/Hour	Average Repair Time (Hours)
OP1	0.001	4.0
OP2	0.005	8.0
OP3	0.005	8.0
OP4	0.005	8.0
OP5	0.005	8.0

Test Suite Characteristics for the Failure based Markov Chain



- Ok state probability – 0.869
- Number of test cases to cover 1.0 of probability mass – 2,002
- Number of test cases to cover 0.99999 of probability mass – 39
- Number of test cases to cover 0.9999 of probability mass – 26
- Number of test cases to cover 0.999 of probability mass – 14
- Number of test cases to cover 0.99 of probability mass – 5

Test Suite Characteristics for the Resource Usage based Markov Chain



- Number of test cases to cover 0.964 of probability mass – 24,492
- Most likely states in the resource usage Markov chain:

N1	N2	N3	N4	N5	~Prob
9	2	2	1	0	0.002
10	2	2	1	0	0.002
9	3	2	1	0	0.002
10	3	2	1	0	0.002
8	2	2	1	0	0.002

Conclusions

- Reliability assessment as a function of mission time that is based on resource usage and failure state test suites
- Assumes resource usage and failure based Markov chains are independent
- Algorithm able to reduce test suite size depending on system reliability objective
- We are currently applying our approach to two large mission critical systems to certify reliability objective and to help make software release decision