Better Reliability Assessment and Prediction Through Data Clustering

TSE 2002
Jeff Tian
Outline

- Introduction
- Overall Approach
- Data Cluster based Reliability Model 1
- Data Cluster based Reliability Model 2
- Evaluation
- Conclusion
- Discussion
Introduction

- Software reliability model
  - A systematic way of assessing and predicting software reliability

Software Reliability Growth Models (SRGMs)

Common assumption
"Failure intervals or period failure counts are assumed to be independent"

Randomized Testing
Introduction

- Testing in a large software system
  - Divided testing efforts into functional area
    - test execution on a smaller, more manageable scale
  - Mixture of scenario-based testing and clustered testing

Scenario based testing
“centered around the framework of scenarios”

Clustered testing
“focused on fault localization & fixing”
Motivation

- Observations in large software systems
  - Short-term dependency among test runs
    - Run correlation due to defect fixing
    - may produce biased results in applying existing SRGMs
  - Long-term independency among test runs
    - Approximating scenario-based testing with random testing
      - Generally conducted by many testers in parallel
    - suggests the possibility of using existing SRGMs
Research Goal

- Propose the better approach to software reliability modeling by grouping data
  - Data Cluster based Reliability Model 1 (DCRM1)
    - A piecewise linear model for short-term reliability assessment and problem identification
  - Data Cluster based Reliability Model 2 (DCRM2)
    - A dual model for long-term reliability assessments and predictions
Overall Approach
(two-staged approach)

(1) Usage dependent time measurement
(2) Data clustering through tree-based modeling
(3) Fitting existing SRGMs to grouped data

Modeling Analysis & Comparison
- Goodness of Fit
- Prediction accuracy
- Stability

DCRM1
- Goel Okumoto model
- Musa Okumoto model

DCRM2
- Existing SRGMs
- Grouped data through DCRM1
Data cluster based reliability model1 (DCRM1)

- Tree-based modeling
  - Statistical analysis technique through recursive partitioning
  - Predictor variable $T_i$: usage time (transaction)
  - Response variable $\lambda_j$: point failure rate

<table>
<thead>
<tr>
<th>$T_i$</th>
<th>$\lambda_j$</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>0</td>
</tr>
<tr>
<td>1000</td>
<td>0</td>
</tr>
<tr>
<td>200</td>
<td>1/200</td>
</tr>
<tr>
<td>500</td>
<td>1/500</td>
</tr>
<tr>
<td>400</td>
<td>1/400</td>
</tr>
<tr>
<td></td>
<td>0</td>
</tr>
</tbody>
</table>

Dataset $S\{\text{Run1, Run2, R3, ...Rn}\}$
Data cluster based reliability model

Step 1. Initialization

- Create a list of datasets to be partitioned, Slist
- Select the size threshold $T_s$
- Select the homogeneity threshold $T_h$
Data cluster based reliability model

Step 2. Overall Control & Size test
- Remove S from Slist
- Repeat next steps until Slist becomes empty
- Size test

1. Initialization
   - Slist
   - $T_s$
   - $T_h$

2. Size Test
   - Remove S
   - $|S| < T_s$
     - Yes $\rightarrow$ STOP
     - No $\rightarrow$ Next step
Data cluster based reliability model 1

Step 3. Defining binary partitions

- Binary partition of $S$: a division of data into two subsets using split condition

![Graph showing binary partition]

- $T < c$
- $T \geq c$
- $S_1$
- $S_2$
- Cutoff value
**Data cluster based reliability model**

Step 4. Computing predicted responses and prediction deviances for S, S1 and S2

- \( \lambda(S_1) \): predicted failure rate for cluster \( S_1 \)
  \[
  \lambda(S_1) = \sum_{i \in S_1} \frac{f_i}{t_i}
  \]

- Prediction deviance
  \[
  D(S_1) = \sum_{i \in S_1} (\lambda_i - \lambda(S_1))^2
  \]

- \( \lambda_i \): point failure rate
Data cluster based reliability model

Step 5. Select the optimal partition
- Select the partition with minimized \( \{D(S_1) + D(S_2)\} \)

Step 6. Homogeneity test
- Check the partitioning improve prediction accuracy beyond a threshold \( T_h \)

1. Initialization

\[
\text{Slist} = \{ S_1, S_2 \}
\]

6. Homogeneity test

\[
(1 - \frac{D(S_1) + D(S_2)}{D(S)}) \leq T_h
\]

Yes

No

STOP
Data cluster based reliability model

Data clusters according to DCRM1

<table>
<thead>
<tr>
<th>segment</th>
<th>$\tau$ cutoff</th>
<th>cumulative failures</th>
<th># of runs</th>
<th>failure rate ($A_i$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>NA</td>
</tr>
<tr>
<td>2</td>
<td>30261536</td>
<td>9</td>
<td>31</td>
<td>2.974e-07</td>
</tr>
<tr>
<td>3</td>
<td>123855436</td>
<td>18</td>
<td>30</td>
<td>9.616e-08</td>
</tr>
<tr>
<td>4</td>
<td>131290212</td>
<td>27</td>
<td>16</td>
<td>1.211e-06</td>
</tr>
<tr>
<td>5</td>
<td>160634323</td>
<td>34</td>
<td>12</td>
<td>2.385e-07</td>
</tr>
<tr>
<td>6</td>
<td>164363318</td>
<td>40</td>
<td>6</td>
<td>1.609e-06</td>
</tr>
<tr>
<td>7</td>
<td>202584007</td>
<td>63</td>
<td>34</td>
<td>6.018e-07</td>
</tr>
</tbody>
</table>

Sn : Segment n

<Data cluster table>
Data cluster based reliability model

- Piecewise linear model
  - Be able to fit to almost any data
  - Predict reliability into the near future
  - Identify trouble spots or problematic areas
    - Problematic test scenarios / Weak product functional areas / Other trouble spots
Weakness of DCRM1

- Negative side
  - Hard to use for long-term reliability prediction
  - Lacks parameters that have meaningful physical interpretations

Extend DCRM1 to compensate for weaknesses of DCRM1
Data cluster based reliability model2 (DCRM2)

Fitting SRGMs to grouped data

Assumptions of grouped data

- Homogeneous testing segments
- Independence among the separate segments

Existing SW Reliability Growth Models

- Goel-Okumoto (GO) model
  
  \[ m(t) = N(1 - e^{-bt}) \]

  - \( N \): estimated total # of defects
  - \( b \): constant for model curvature

- Musa-Okumoto (MO) model
  
  \[ m(t) = \frac{1}{\theta} \log(\lambda_0 \theta t + 1) \]

  - \( \lambda_0 \): estimated initial failure rate
  - \( \theta \): constant for model curvature

\[ \text{Grouped data} \]

Modeling Result Summary:

- \( m(t) = N(1 - \exp(b't)) \)
- \( N \): 133
- \( b \): 2.9498e-09
- \( \text{failure rate: } 8.203e-09 \)
- \( \text{MTBF: } 121900288 \)
- \( \text{P: } 3907 \)
Modeling Result Analysis and Comparison

Data description

- Transactions for each test run
- Time-stamp for each test run
- Related failure observations

Comparison models

1. DCRM1
2. Existing SRGMs (GO and MO models)
   - fitted to the raw data from products D and E
3. DCRM2.GO and DCRM2.MO
   - fitted to data clusters identified by DCRM1
Modeling applicability and points for comparison

- Data description
  - Weekly modeling activities

<table>
<thead>
<tr>
<th>Week</th>
<th>F-7</th>
<th>F-6</th>
<th>F-5</th>
<th>F-4</th>
<th>F-3</th>
<th>F-2</th>
<th>F-1</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cumulative failures</td>
<td>102</td>
<td>104</td>
<td>109</td>
<td>116</td>
<td>122</td>
<td>125</td>
<td>128</td>
<td>128</td>
</tr>
<tr>
<td>Cumulative transactions</td>
<td>4.58X10^8</td>
<td>5.43X10^8</td>
<td>6.14X10^8</td>
<td>7.01X10^8</td>
<td>9.77X10^8</td>
<td>1.08X10^8</td>
<td>1.25X10^8</td>
<td>1.32X10^8</td>
</tr>
</tbody>
</table>
Goodness-of-fit comparison

- Goodness-of-fit measure
  - Sum of residual squares

\[ R^2 = \sum_i (f_i - \hat{f}_i)^2 \]

- \( f_i \): cumulative failures for i data point
- \( \hat{f}_i \): predicted cumulative failures by model

<table>
<thead>
<tr>
<th>week</th>
<th>( F - 7 )</th>
<th>( F - 6 )</th>
<th>( F - 5 )</th>
<th>( F - 4 )</th>
<th>( F - 3 )</th>
<th>( F - 2 )</th>
<th>( F - 1 )</th>
<th>( F )</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCRM1</td>
<td>382</td>
<td>390</td>
<td>456</td>
<td>443</td>
<td>453</td>
<td>521</td>
<td>582</td>
<td>566</td>
</tr>
<tr>
<td>GO (raw)</td>
<td>12971</td>
<td>17891</td>
<td>22072</td>
<td>24483</td>
<td>25773</td>
<td>26723</td>
<td>26871</td>
<td>26925</td>
</tr>
<tr>
<td>DCRM2.GO</td>
<td>14260</td>
<td>22178</td>
<td>24588</td>
<td>25743</td>
<td>26767</td>
<td>27563</td>
<td>27626</td>
<td>27662</td>
</tr>
<tr>
<td>MO (raw)</td>
<td>13061</td>
<td>18784</td>
<td>25006</td>
<td>30366</td>
<td>36254</td>
<td>44201</td>
<td>48483</td>
<td>50794</td>
</tr>
<tr>
<td>DCRM2.MO</td>
<td>14401</td>
<td>24218</td>
<td>28640</td>
<td>32119</td>
<td>37459</td>
<td>45065</td>
<td>50116</td>
<td>53196</td>
</tr>
</tbody>
</table>

<Goodness-of-fit values for different models>

**DCRM1 are significantly smaller** than those for SRGMs
Differences bw **SRGMs and DCRM2 are similar** with about 5%
## Prediction accuracy comparison

### Prediction error

<table>
<thead>
<tr>
<th>Model week ($W_1$)</th>
<th>$F - 7$</th>
<th>$F - 6$</th>
<th>$F - 5$</th>
<th>$F - 4$</th>
<th>$F - 3$</th>
<th>$F - 2$</th>
<th>$F - 1$</th>
<th>$F$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$F - 7$</td>
<td>0</td>
<td>16.7</td>
<td>30.1</td>
<td>45.8</td>
<td>89.6</td>
<td>103.6</td>
<td>125.5</td>
<td>133.4</td>
</tr>
<tr>
<td>$F - 6$</td>
<td>NA</td>
<td>0</td>
<td>7.2</td>
<td>14.7</td>
<td>31.2</td>
<td>35.3</td>
<td>40.5</td>
<td>42.1</td>
</tr>
<tr>
<td>$F - 5$</td>
<td>NA</td>
<td>NA</td>
<td>0</td>
<td>6.57</td>
<td>20.3</td>
<td>23.5</td>
<td>27.4</td>
<td>28.6</td>
</tr>
<tr>
<td>$F - 4$</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>0</td>
<td>12.7</td>
<td>15.6</td>
<td>19.1</td>
<td>20.2</td>
</tr>
<tr>
<td>$F - 3$</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>0</td>
<td>2.46</td>
<td>5.42</td>
<td>6.27</td>
</tr>
<tr>
<td>$F - 2$</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>0</td>
<td>2.60</td>
<td>3.33</td>
</tr>
<tr>
<td>$F - 1$</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>0</td>
<td>0.68</td>
</tr>
<tr>
<td>$F$</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>0</td>
</tr>
</tbody>
</table>

*Error in predicted additional failures by DCRM2.GO*
Prediction accuracy comparison

Prediction error

DCRM2 consistently performs better or equally to SRGM on the raw data.
Prediction stability comparison

- Stability of parameters estimated from data

- DCRM2 shows the better stability

**Model parameter N over time**

**Model parameter b over time**
Model sensitivity to segment size

Criteria for model selection accuracy

- $T_s$: lower bound on the length of the partitioned segments
- Lower $T_s$ yields more data clusters
- Recommend using $T_s = 10$
  - Good performance results for DCRM2
  - Effective usage of DCRM1 to analyze local variations
Conclusion

- Consider short-term dependency and long-term independency among test runs observed in large software systems.

- Develop data cluster based reliability models
  - DCRM1
    - Fit actual failure data better than traditional SRGMs
    - Help us evaluate product reliability in short-term
    - Identify anomalies during the early part of testing
  - DCRM2
    - Consistently outperformed corresponding SRGMs in prediction accuracy and model stability
Pros & Cons

Pros

- The first application of a data grouping idea into SW reliability modeling
- Suggest the model applicable in the early phase of testing

Cons

- Group neighboring data points with homogeneous failure rates
  - Require content-based run dependency analysis
- Require too much data for clustering
  - # of transactions per test run
- Thank you -

Question and Answer
Introduction

Motivation

- Good fitness does not mean the good prediction.
- A project manager is interested in future behavior of software rather than good fitness of past data.
- It is important to select the model in the perspective of predictive ability.

Current time

# of cumulative failures

Model 1
Model 2

Model 2 shows the better prediction!

- Observed data

Model 1 is selected!
**Introduction**

- We observe the following
  - Overall testing process resembles random testing -> long term independence -> suggests the possibility of using properly treated data with existing SRGMs
  - There may be dependencies among test runs within a short time window -> short term dependency -> existing SRGMs fitted to raw data may produce biased results because period independence assumption is violated
Data cluster based reliability model

- Basic assumptions
  - Clustering periods with homogeneous failure intensities
  - Satisfied under the testing environment for many large software systems