Software Reliability Growth Models: Systematic Descriptions and Implementations

I. Corro Ramos¹

E. Brandt², A. Di Bucchianico¹ and R. Henzen²

ENBIS7 Dortmund, Germany, September 26th, 2007

¹ Eindhoven University of Technology, ² Refis System Reliability Engineering
Outline

Introduction

Systematic description of software reliability growth models

Systematic implementations: software reliability tool

Conclusions

Questions
Windows Boot Error

Windows Boot Manager has experienced a problem.

Status: 0xc000000f

Info: An error occurred transferring execution.

You can try to recover the system with the Microsoft Windows System Recovery Tools. (You might need to restart the system manually.)

If the problem continues, please contact your system administrator or computer manufacturer.
Software testing
Reliability growth data

Goel-Okumoto and Yamada S-shaped models

Expected number of failures

Time

/ department of mathematics and computer science
Software reliability growth models (SRGM)

Statistics is needed as exhaustive testing is not always feasible.

SRGM used to:
- Resource planning
- Release decisions
- Certification
- Find one or two more (QE paper)

Typical statistical quantities of interest:
- Probability of no failure in a given time period
- Expected time until next failure
- Number of errors left
Problems

- Insufficient documentation of the models
- More than 200 models known
- Out-dated algorithms

What is needed:

- Systematic description of SRGM (general methodology + algorithms)
- Support pre-selection of models
- Apply specific state-of-the-art algorithms for the models (convergence issues)
- Tool to perform analyses
Systematic description of software reliability growth models

1. Probability model
2. Trend analysis
3. Model estimation
4. Model validation
5. Prediction
Systematic description of software reliability growth models (Cont.)

1. Probability model
   1.1 Joint distribution of \( \{N(t)\}_{t>0} \) and/or \( T_1, T_2, \ldots, T_n \) and/or \( X_1, X_2, \ldots, X_n \) (including (in)dependence structure)
   1.2 Model assumptions
   1.3 Interpretation of model parameters

2. Trend analysis
   2.1 ...
Ungrouped or exact data

\[ X_0 \quad X_1 \quad X_2 \quad X_3 \quad X_4 \]

\[ T_0 = 0 \quad T_1 \quad T_2 \quad T_3 \quad T_4 \quad T_5 \]
Grouped or interval count data
SRGM description

- **GOS** models: $T_1, T_2, \ldots, T_n$ order statistics of a sample $Z_1, Z_2, \ldots, Z_N$ with c.d.f. $F_\theta$.
  - Jelinski-Moranda or EOS: $X_i \sim \text{Exp}(\lambda(N - i + 1))$ i.i.d.

- **NHPP** models: $N(t_i) - N(t_{i-1}) \sim \text{Poisson}(\lambda(t))$ independent with mean

\[
\mathbb{P}[N(t_i) - N(t_{i-1}) = k] = e^{-(\Lambda(t_i) - \Lambda(t_{i-1}))} \frac{\left(\Lambda(t_i) - \Lambda(t_{i-1})\right)^k}{k}
\]

- Goel-Okumoto: $\Lambda(t) = a(1 - e^{-bt})$
- Yamada S-shaped: $\Lambda(t) = a(1 - (1 + bt)e^{-bt})$
- Duane: $\Lambda(t) = \left(\frac{t}{\alpha}\right)^\beta$
3. Parameter estimation

3.1 Point estimation procedures for parameters (Maximum Likelihood and/or Least Squares)

3.1.1 Data requirements
3.1.2 Existence results for parameter estimates
3.1.3 Performance of parameter estimators (bias, efficiency)
3.1.4 Algorithms to compute parameter estimates

3.2 Confidence interval procedures for parameters (Maximum Likelihood and/or Least Squares)

3.2.1 Distributional description of underlying estimators
3.2.2 Algorithms to compute confidence intervals
Model estimation

- The unknown number of errors $N$ is also a parameter (standard asymptotic theory is not applicable)
- ML equations usually do not have closed-form solution (use specific algorithm for each model, see Knafl and Morgan for some NHPP models)
- Direct maximization may cause numerical problems (parameters of different order of magnitude (Yin, Trivedi paper))
## Model (pre-)selection

<table>
<thead>
<tr>
<th>Data Requirements and Assumptions</th>
<th>Relative Importance</th>
<th>Jelinski-Moranda</th>
<th>Littlewood-Verral</th>
<th>Musa basic</th>
<th>Musa-Okumoto</th>
<th>Goel-Okumoto</th>
<th>Shich-Wolverton</th>
<th>Schneidwind</th>
<th>Yamada S-shaped</th>
<th>Duane</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data may be exact failure times (ungrouped data)</td>
<td>2</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data may be grouped failure times (interval count data)</td>
<td>2</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Testing intervals may be of different length</td>
<td>3</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Failures need not occur equally likely</td>
<td>2</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detection of faults may be dependent of each other</td>
<td>2</td>
<td>x</td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Failures need not be of the same severity</td>
<td>3</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Detection rate depends on time (testing effort)</td>
<td>3</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detection rate depends on number of remaining defects</td>
<td>3</td>
<td>x</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Failures need not be repaired instantaneously</td>
<td>3</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>Imperfect repair of defects allowed</td>
<td>2</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Infinite number of errors allowed</td>
<td>2</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Systematic implementations: software reliability tool

- Existing packages do not make full use of state-of-the art statistical methodology
- Interface in Java (platform independent)
- Statistical computations in $\mathbb{R}$ (open-source free software)
- Communication: JRI and JavaGD libraries (Computational Statistics group, University of Augsburg)
- Financially supported by a grant of the Dutch Innovation Platform
Example
<table>
<thead>
<tr>
<th>Observation Number</th>
<th>times</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>39</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>36</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>91</td>
</tr>
<tr>
<td>9</td>
<td>49</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>25</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>30</td>
</tr>
<tr>
<td>15</td>
<td>42</td>
</tr>
<tr>
<td>16</td>
<td>9</td>
</tr>
<tr>
<td>17</td>
<td>49</td>
</tr>
<tr>
<td>18</td>
<td>44</td>
</tr>
<tr>
<td>19</td>
<td>32</td>
</tr>
<tr>
<td>20</td>
<td>3</td>
</tr>
<tr>
<td>21</td>
<td>78</td>
</tr>
<tr>
<td>22</td>
<td>1</td>
</tr>
<tr>
<td>23</td>
<td>30</td>
</tr>
<tr>
<td>24</td>
<td>205</td>
</tr>
<tr>
<td>25</td>
<td>5</td>
</tr>
</tbody>
</table>
Results Laplace Trend Test

- Data set: Joe_J data
- Number of observations: 207
- Variables: Cumulative Times
- dH0: 1%
- Test statistic: 6.672
- Lower Critical Value: -2.776
- Upper Critical Value: 2.776
- P-Value: 0.000

Conclusion:
The test statistic is small, which indicates that there is no significant growth in reliability.

Results Military Handbook 189 Trend Test

- Data set: Joe_J data
- Number of observations: 207
- Variables: Cumulative Times
- dH0: 1%
- Test statistic: 611.941

Graphical Output
Results Laplace Trend Test

Data set: Joe_L.xls
Number of observations: 207
Variable: Cumulative Times
α: 1%
Test statistic: 6.67231
Lower Critical Value: -2.57583
Upper Critical Value: 2.57583
P-Value: 0.00000

Conclusion
The test statistic is small, which indicates that there is a significant growth in reliability.

Results Military Handbook 189 Trend Test

Data set: Joe_L.xls
Number of observations: 207
Variable: Cumulative Times
α: 1%
Test statistic: 611.94130
Lower Critical Value: 341.81800
Upper Critical Value: 489.69129
P-Value: 0.00000
Model Select Wizard

Assumptions:
- Data may be exact failure times (ungrouped data)
- Data may be grouped failure times (interval count data)
- Testing intervals may be of different length
- Failures need not occur equally likely
- Detection of faults may be dependent of each other
- Failures need not be of the same severity
- Detection rate depends on time (testing effort)
- Detection rate depends on number of remaining defects
- Failures need not be repaired instantaneously
- Imperfect repair of defects allowed
- Infinite number of errors allowed

Preferred models:  
- Goel-Okumoto: 83%  
- Yamada S-shaped: 83%  
- Musa basic: 81%  
- Schneidewind: 80%  
- Littlewood-Verrall: 80%  
- Jelinski-Moranda: 77%  
- Shick-Wolverton: 77%  
- Geometric: 75%  
- Duane: 75%  
- Musa-Okumoto: 71%
Analysis Output

Dataset: Joe_Lids
Nr. of Observations: 207
Variable: Cumulative Loss
&alpha: 1%

Model: Goel-Okumoto

Model estimates:

a = 255.42933
b = 0.00010

Kolmogorov-Smirnov Goodness of Fit Test:
Conclusions

- Systematic description of software reliability growth models needed
- Model assumptions
- Numerical problems in model estimation
- Results for both grouped and ungrouped data
- Software reliability tool
- Programmed in Java (interface) and \( \mathbb{R} \) (statistical computations)
Questions
Advanced statistical issues

- Model validation
  - Standard goodness-of-fit test result inappropriate in general
  - Alternatively: conditional goodness-of-fit tests
  - Unconditional goodness-of-fit tests for diverse subclasses of models, only for ungrouped data!

- Model prediction
  - One-sided confidence intervals
  - Asymptotic confidence intervals ($N \to \infty$ or $t \to \infty$)
  - Number of remaining errors, reliability (time to next error) of the system, predicted intensity function, etc...