SENG 637
Dependability, Reliability & Testing of Software Systems

Software Reliability Tools
(Chapter 8)

Department of Electrical & Computer Engineering, University of Calgary

B.H. Far (far@ucalgary.ca)
http://www.enel.ucalgary.ca/People/far/Lectures/SENG637/
SRE: Process (Review)

- 5 steps in SRE process:
  - Define necessary reliability
  - Develop operational profiles
  - Prepare for test
  - Execute test
  - Apply failure data to guide decisions
Chapter 8 Part 1

Software Reliability

Engineering Tools
SRE Tools: Reliability Growth

**Failure data**

<table>
<thead>
<tr>
<th>Time(s)</th>
<th>Cumulative Failures</th>
<th>Failures in interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>60</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>90</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>120</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>150</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>180</td>
<td>11</td>
<td>1</td>
</tr>
<tr>
<td>210</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>240</td>
<td>13</td>
<td>1</td>
</tr>
<tr>
<td>270</td>
<td>14</td>
<td>1</td>
</tr>
</tbody>
</table>

**Output data**

**Failure Intensity**

far@ucalgary.ca
Tasks Handled by SRE Tools

- Collecting failure and test time information
- Calculating estimates of model parameters using this information
- Testing to fit a model against the collected information
- Selecting a model to make predictions of remaining faults, time to test, etc.
- Applying the model
Available Options

Selection of a tool is one of the important decisions in performing the SRE study.

An inappropriate choice may not handle the type of data collected for the project, or does not have a robust set of models that may fit to the project to make accurate predictions of important information.

Engineers may choose between:

- Using a general-purpose application program such as a spreadsheet or a statistical package such as SAS and developing their own models using a general-purpose programming language such as JAVA or C.
- Using a shareware, freeware or commercially available SRE tool.
All of the SRE tools use one of two basic types of input data:

- time-domain data (i.e., time-between-failures data)
- interval-domain data (i.e., failure-count data)
Input Data Specification

1) Time of failure
2) Time interval between failures
3) Cumulative failure up to a given time
4) Failures experienced in a time interval

<table>
<thead>
<tr>
<th>Failure no.</th>
<th>Failure times (hours)</th>
<th>Failure interval (hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>2</td>
<td>19</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>32</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>43</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>58</td>
<td>15</td>
</tr>
<tr>
<td>6</td>
<td>70</td>
<td>12</td>
</tr>
<tr>
<td>7</td>
<td>88</td>
<td>18</td>
</tr>
<tr>
<td>8</td>
<td>103</td>
<td>15</td>
</tr>
<tr>
<td>9</td>
<td>125</td>
<td>22</td>
</tr>
<tr>
<td>10</td>
<td>150</td>
<td>25</td>
</tr>
<tr>
<td>11</td>
<td>169</td>
<td>19</td>
</tr>
<tr>
<td>12</td>
<td>199</td>
<td>30</td>
</tr>
<tr>
<td>13</td>
<td>231</td>
<td>32</td>
</tr>
<tr>
<td>14</td>
<td>256</td>
<td>25</td>
</tr>
<tr>
<td>15</td>
<td>296</td>
<td>40</td>
</tr>
</tbody>
</table>
Input Data Specification

1) Time of failure
2) Time interval between failures
3) Cumulative failure up to a given time
4) Failures experienced in a time interval

<table>
<thead>
<tr>
<th>Time(s)</th>
<th>Cumulative Failures</th>
<th>Failures in interval</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>60</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>90</td>
<td>7</td>
<td>2</td>
</tr>
<tr>
<td>120</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>150</td>
<td>10</td>
<td>2</td>
</tr>
<tr>
<td>180</td>
<td>11</td>
<td>1</td>
</tr>
<tr>
<td>210</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>240</td>
<td>13</td>
<td>1</td>
</tr>
<tr>
<td>270</td>
<td>14</td>
<td>1</td>
</tr>
</tbody>
</table>
SRE Tools

- CASRE
- SoftRel
- SMERFS
- SoRel
- SRMP
- ProConf
- Relex
- MEADEP
  (MEAsure and DEPendability)

SRE tool repository:

1. Open channel software:
   http://www.openchannelsoftware.org/discipline/Reliability_Analysis/

2. Univ of Maryland
   http://www.enre.umd.edu/tool.htm

etc.
SRE Tools (cont’d)

- **ACARA II:** Availability, Cost, And Resource Allocation, Version 2 (no charge per license)
- **ARAM:** Automated Reliability/Availability/Maintainability, Version 2.0 ($400 source code license)
- **ETARA:** Event Time Availability, Reliability Analysis ($200 source code license)
- **GO:** Graphics Oriented Program ($150 source code license)
- **HARP:** Hybrid Automated Reliability Predictor, Version 7.0 ($500 source code license, for Unix or PC)
- **HARPO:** Hybrid Automated Reliability Predictor Output Graphics Display ($150 source code license)
- **SPRPM:** Software Problem Report Metrics Program (no charge per license, requires EXCEL)
More Info

- Download tools:
  - IEEE Software Reliability Engineering Working Group (SREWG)
    http://www.srewg.org/Tools/

- SRE tools repository:
  - Center for Reliability Engineering at the University of Maryland
    http://www.enre.umd.edu/tool.htm

- Open Channel Foundation
  http://www.openchannelsoftware.org/discipline/Reliability_Analysis
Chapter 8 Section 2

How to use CASRE
CASRE: Introduction

- Software Reliability Estimation tool running on Windows
- CASRE extends the SMERFS package by adding a menu based GUI
- Uses ASCII text input data files
- Displays results in tabular and/or graphical form
- Can use many different models
CASRE Program Structure

- **Main Window**
  - The window where the input data file is loaded and displayed.
  - Menu options allow the user to apply models and filters to the input data.

- **Graphical Display Window**
  - Displays a plot of the input data, as well as the results of any models applied to the data.

- **Model Results Table**
  - Displays the tabulated results from the models that were used in the calculation.
Main Window

- The main window is the starting point for CASRE sessions. This is the place where the user selects the models and filters to apply to the input data.

- Menu Options
  - File (Open, Save, Print, Exit)
  - Edit (Change Data Type, External Application, Escape to DOS)
  - Filters (Shaping and Scaling, Change time unit, etc.)
  - Model (Select and Run, Define Combination, Edit/Remove Models, Parameter Estimation, Select Data Range, Predictions)
  - Setup, Plot, Help
Graphical Display

- Provides the plots of the input and calculated data.
- Each individual data set on a plot has its own unique symbol and colour.
- Menu Options
  - Plot (Save as, Draw from File, Setup Printer, Print Plot)
  - Results (Select Model Results, Model Results Table)
  - Display
    - Graphs - Time between failures, Failure counts, Failure intensity, Test interval lengths, Cumulative failures, Reliability
    - Model Evaluation - Goodness-of-fit, Prequential likelihood, Relative accuracy, Bias, Bias trend, Bias scatter plot, Model noise, Model ranking
  - Settings, Copy, Help
Model Results Window

- Displays the detailed calculated results in a tabular format.
- Reliability estimates, parameter estimates, and convergence information are all displayed in this table for a selected model.

Menu Options
- File
- Results – Select Results, Previous Model, Next Model
- Help

<table>
<thead>
<tr>
<th>Test interval</th>
<th>Failures per test interval</th>
<th>Test interval Length - Hours</th>
<th>Next step Prediction</th>
<th>Model Estimates</th>
<th>Est. reliability: 5.6000e+001 Hr</th>
<th>Est'd cumulative number of failures</th>
<th>Estimated failure intensity at T</th>
<th>Actuats-Estimates</th>
<th>Did Estimated Converge?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.60000e+001</td>
<td>5.60000e+001</td>
<td>N/A</td>
<td>2.33054e+001</td>
<td>7.53086e-011</td>
<td>2.33054e+001</td>
<td>4.16238e+001</td>
<td>-9.30841e+000</td>
<td>N/A</td>
</tr>
<tr>
<td>2</td>
<td>1.50000e+001</td>
<td>5.60000e+001</td>
<td>N/A</td>
<td>2.11500e+001</td>
<td>3.56391e-010</td>
<td>4.40358e+001</td>
<td>3.38402e+001</td>
<td>-2.15493e+000</td>
<td>N/A</td>
</tr>
<tr>
<td>3</td>
<td>2.30000e+001</td>
<td>5.60000e+001</td>
<td>N/A</td>
<td>1.86434e+001</td>
<td>2.33634e-009</td>
<td>6.24501e+001</td>
<td>3.5081e-001</td>
<td>3.35458e+000</td>
<td>N/A</td>
</tr>
</tbody>
</table>
CASRE Data Input

- ASCII based text file with a .dat extension
- Two file formats
  - Time Between Failures (error #, time since last failure, failure severity class)
  - Failure Counts (interval #, # errors in interval, interval length, failure severity class)
- The format of the file must be strictly adhered to
- No direct manipulation of the data file is allowed but CASRE has menu links to common text editors
Using CASRE /1

1. Prepare input data
   - Input data can be either failure count or failure per interval data

Sample failure count data

| 1  | 30  | 1 |
| 2  | 55  | 1 |
| 3  | 70  | 1 |
| 4  | 60  | 1 |
| 5  | 90  | 1 |
| 6  | 110 | 1 |
| 7  | 100 | 1 |
| 8  | 150 | 1 |
| 9  | 120 | 1 |
| 10 | 215 | 1 |
1. Prepare input data

- Input data can be either failure count or failure per interval data

Sample failure per interval data

<table>
<thead>
<tr>
<th>&lt;interval number&gt;</th>
<th>&lt;failure in interval&gt;</th>
<th>&lt;duration of interval&gt;</th>
<th>&lt;severity class&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>2.5</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>3</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>2</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>1.5</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>3</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>4</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>2.5</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>3</td>
<td>0 0 0 1</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>5</td>
<td>0 0 0 1</td>
</tr>
</tbody>
</table>
Using CASRE /3

2. Check if data shows reliability growth (trend test)
3. Read input file
4. Select data range
5. Filter or smooth input data if required
6. Select parameter estimation method
7. Select and run model(s)
8. View and interpret model results
   - Goodness of fit test
   - Model ranking
   - Prediction based on plots
- Time between failure models
  - Geometric
  - Jelinski-Moranda
  - Littlewood-Verrall
  - Musa-Basic
  - Musa-Okumoto
  - NHPP
CASRE Reliability Models /2

- Failure Count models
  - Generalized Poisson
  - NHPP
  - Schneidewind
  - Shick-Wolverton
  - Yamada S-shaped
CASRE Reliability Models

Combination models:

- Four predefined models
  - Dynamically weighted
  - Equally weighted
  - Median Weighted
  - Unequally weighted
- Other combination models can be defined
Trend Check

- CASRE models should only be used on data where the overall reliability is increasing as testing continues.
- Reliability is increasing if the mean time between failures increases as the total number of failures increases during testing.
- CASRE version 2 does not have the automatic trend test option, so a visual inspection of the cumulative failure plot is necessary.
- CASRE version 3 has an automatic trend test option, that will inform the user if the data is applicable to the reliability models.
Trend Related Questions ...

- Is the system reliability increasing, decreasing or stable?
- Which reliability growth model fits best the gathered data?
- Can the same model be used in all cases of reliability growth, decrease and stable?
And Trend Related Answers ...

- Reliability trends can be analyzed by “trend tests”.

- Trend tests can be used to help determine whether the system undergoes reliability growth, decrease or stable reliability.

- Trend analysis also helps select appropriate reliability model for each phase.
Failure Data for Trend Tests

- The trend tests work with the failure data.
- The trend can be analyzed using
  - Inter-failure times data or
  - Failure intensity data
Two trend tests are commonly carried:

- Arithmetical mean test
- Laplace tests
The arithmetical mean of the inter-failure times consists of calculating arithmetical mean $\tau(i)$ of the observed inter-failure times $\theta_j$.

$$
\tau(i) = \frac{1}{i} \sum_{j=1}^{i} \theta_j
$$

An increasing series of $\tau(i)$ indicates reliability growth and a decreasing series suggests reliability decrease.
Inter-failure Times Data

For $N(T)$ as the cumulative number of failures over the time period $[0, T]$, the Laplace factor $u(T)$ is derived:

$$u(i) = \frac{1}{i-1} \sum_{n=1}^{i-1} \sum_{j=1}^{n} \theta_j - \frac{1}{2} \sum_{j=1}^{i} \theta_j \sqrt{\frac{1}{12(i-1)}}$$

For the case that $T$ is equal to the time of occurrence of failure $i$. 

far@ucalgary.ca
Inter-failure Times Data

- Negative values of the Laplace factor $u(i)$ indicate a decreasing failure intensity, i.e., reliability growth.
- Positive values of the Laplace factor $u(i)$ indicate an increasing failure intensity, i.e., reliability decrease.
- Values between $-2$ and $+2$ indicate stable reliability.
Failure Intensity Data

- For the time period $[0, T]$, divided into $k$ units of equal length and for $n(i)$ be the number of failures observed during the time interval $i$, the Laplace factor $u(k)$ is derived by:

$$u(k) = \frac{\sum_{i=1}^{k} (i-1)n(i) - \left(\frac{k-1}{2}\right)\sum_{i=1}^{k} n(i)}{\sqrt{\frac{k^2 - 1}{12} \sum_{i=1}^{k} n(i)}}$$
Failure Intensity Data

- Negative values of the Laplace factor $u(k)$ indicate a decreasing failure intensity, i.e., reliability growth.
- Positive values of the Laplace factor $u(k)$ indicate an increasing failure intensity, i.e., reliability decrease.
Typical Plots

- Typical graphs for failure intensity $n(k)$ and cumulative failure intensity $N(k)$
Typical plot for the Laplace factor $u(k)$
Typical Plots /3

- Typical plot for Laplace factor during various project phases

Decrease of reliability

Reliability growth

u(k)

Validation  Field test  Operation

k
Selecting Models

- Typical plot for Laplace factor during various project phases

Decrease of reliability
Only models allowing increasing failure intensity can be applied

Reliability growth (any reliability growth model can be applied)

Reliability growth
CASRE: Case Study

- Project X is a web based application for accessing a database using a browser.
- This version of the software is a minor release with changes to the GUI display and data access engine.
- Two programmers were assigned to the project. One programmer worked on the GUI, and the other on the data access engine.
- The project took approximately 4 weeks to complete.
Case Study (contd.)

- A single tester was assigned to the project.
- The test phase was completed in approximately 25 hours (3 working days or 90,000 seconds).
- 136 failures were discovered during the testing.
- Using the dates and times recorded for the failures discovered during testing, a “time between failures” input file was generated for CASRE.
- The severity of all the failures was set to
  - 1 - Low Severity
Time Between Failures Plot
Trend Analysis

- Laplace test shows reliability growth.
Project Results

In order to determine which models would provide the best fit for the project data, the following models were run:

- Geometric
- Jelinski - Moranda
- Littlewood - Verrall
- Musa Basic
- Musa - Okumoto
Goodness of Fit Test

On Graphic display window select:
Display → Goodness of fit

<table>
<thead>
<tr>
<th>Model Name</th>
<th>KS Distance</th>
<th>95% Fit?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geometric</td>
<td>8.811010e-002</td>
<td>Yes</td>
</tr>
<tr>
<td>Jelinski-Moranda</td>
<td>9.443151e-002</td>
<td>Yes</td>
</tr>
<tr>
<td>Littlewood-Verrall</td>
<td>7.504933e-002</td>
<td>Yes</td>
</tr>
<tr>
<td>Musa Basic</td>
<td>9.203077e-002</td>
<td>Yes</td>
</tr>
<tr>
<td>Musa-Okumoto</td>
<td>8.791571e-002</td>
<td>Yes</td>
</tr>
</tbody>
</table>
On Graphic display window select:
Display → Model rankings → Rank summary or Rank details

<table>
<thead>
<tr>
<th>Model Name</th>
<th>Rank</th>
<th>Reliability</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geometric</td>
<td>1</td>
<td>1.463655e-001</td>
</tr>
<tr>
<td>Littlewood-Verrall</td>
<td>1</td>
<td>9.667713e-002</td>
</tr>
<tr>
<td>Musa-Okumoto</td>
<td>1</td>
<td>1.576047e-001</td>
</tr>
<tr>
<td>Musa Basic</td>
<td>4</td>
<td>3.276631e-001</td>
</tr>
<tr>
<td>Jelinski-Moranda</td>
<td>5</td>
<td>3.702861e-001</td>
</tr>
</tbody>
</table>
Display Results

On Graphic display window select: Results → Select model results

Select and Display Model Results

- Models for which predictions cannot be displayed

- Select Model Results to Display
  - Models executed
    - Jelinski-Moranda
    - Musa Basic

- Results to display
  - Geometric
  - Littlewood-Verrall
  - Musa-Okumoto

Only 3 graphs can be displayed at a time
Display: Cumulative Failures
Display: Time Between Failures
Interpreting Results

- Accuracy of estimation of the failure intensity $\lambda$ depends on the number of failures experienced (i.e., the sample size).
- Good results in estimating failure intensity are generally experienced for programs with 5,000 or more developed source lines.
- Satisfactory results are obtained for programs with 1,000 or more developed source lines.
How to Handle Defects?

- Table below gives the time between failures for a software system:

<table>
<thead>
<tr>
<th>Error no.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time since last failure (hours)</td>
<td>6</td>
<td>4</td>
<td>8</td>
<td>5</td>
<td>6</td>
<td>9</td>
<td>11</td>
<td>14</td>
<td>16</td>
<td>19</td>
</tr>
</tbody>
</table>

- What can we learn from this data?
  - system reliability?
  - total number of errors in the system?
  - time to (approximately) remove all errors?
What to Learn from Data?

The inverses of the inter-error times are the failure intensity data points (or error rate if plotted against time).

<table>
<thead>
<tr>
<th>Error no.</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time since last failure (hours)</td>
<td>6</td>
<td>4</td>
<td>8</td>
<td>5</td>
<td>6</td>
<td>9</td>
<td>11</td>
<td>14</td>
<td>16</td>
<td>19</td>
</tr>
<tr>
<td>Failure intensity</td>
<td>0.166</td>
<td>0.25</td>
<td>0.125</td>
<td>0.20</td>
<td>0.166</td>
<td>0.111</td>
<td>0.09</td>
<td>0.071</td>
<td>0.062</td>
<td>0.053</td>
</tr>
</tbody>
</table>
Interpreting Results /2

- When the failure intensity $\lambda$ is very large and the trend indicates little chance of achieving the $\lambda_F$ by the scheduled release date, what can be done?
  - Adding additional test and debugging resources
  - Adjusting the balance among the objectives for failure intensity, development time, and development cost
  - Deferring features
Conclusions

- CASRE is a valuable tool for software reliability estimation.
- CASRE is easy to learn and use. It is possible to become proficient at the software in a few hours.
- For accuracy in the calculated results, CASRE should be applied to projects where the expected number of failures is greater than 40 to 50 failures.
Software Release!

far@ucalgary.ca