Reliability Measurement Goal

- Reliability measurement is a set of mathematical techniques that can be used to estimate and predict the reliability behavior of software during its development and operation.

- The primary goal of software reliability modeling is to answer the following question:

  Given a system, what is the probability that it will fail in a given time interval, or, what is the expected duration between successive failure?
Basic Definitions

- **Software Reliability** $R(t)$: The probability of failure-free operation of a computer program for a specified time under a specified environment.

- **Failure**: The departure of program operation from user requirements.

- **Fault**: A defect in a program that may cause a failure.

- **Error**: Human action that results in software containing a fault.
Basic Definitions

- **Failure Intensity (rate) $f(t)$**: The expected number of failures experienced in a given time interval.

- **Mean-Time-To-Failure (MTTF)**: Expected value of a failure interval.

- **Expected total failures $u(t)$**: The number of failures expected in a time period $t$. 
Reliability Theory

Let "t" be a random variable representing the failure time or lifetime of a physical system.

For this system, the probability that it will fail by time "t" is:

\[ F(t) = P[T \leq t] = \int_0^t f(x) \, dx \]

The probability of the system surviving until time "t" is:

\[ R(t) = P[T > t] = 1 - F(t) = \int_t^\infty f(x) \, dx \]
Reliability Theory

A reliability objective expressed in terms of one reliability measure can be easily converted into another measure as follows (assuming an “average” failure rate, $\lambda$, is measured):

\[
\begin{align*}
  u(t) &= \lambda \times t \\
  MTTF &= \frac{1}{\lambda} \\
  \lambda &= \frac{1}{MTTF} \\
  R(t) &= \exp(-u(t));
\end{align*}
\]
Basic Execution Time Model

\[ \lambda(u) = \lambda_0 \left[ 1 - \frac{u}{v_0} \right] \]

\[ u(t) = v_0 \left[ 1 - \exp\left[ -\frac{\lambda_0}{v_0} t \right] \right] \]

\[ \lambda(t) = \lambda_0 \exp\left[ -\frac{\lambda_0}{v_0} t \right] \]
Examples

1. Assume that a program will experience 100 failures in infinite time. It has now experienced 50. The initial failure intensity was 10 failures/CPU hr. The current failure intensity is:

2. A program with an initial failure intensity of 10 failures/CPU hr and 100 total failures. We will look at the failure experienced after 10 and 100 CPU hr of execution.
Logarithmic Poisson Model

\[ \lambda(u) = \lambda_0 e^{-\theta u} \]
\[ u(t) = \frac{1}{\theta} \ln(\lambda_0 \theta t + 1) \]
\[ \lambda(t) = \frac{\lambda_0}{\lambda_0 \theta t + 1} \]
Availability

- Fraction of time the system is not OFF.
- Fraction of TOTAL TIME the system is not DOWN.
- Fraction of UP TIME the system is OPERATING.
- Represents: probability that a given function can be invoked by the user at a specified time.
Maintainability

• Represents: probability that a given function can be made available (within time \( t \)) given that it is not currently available.

• Mean time to Restore

• Mean time to Repair

• Expected Down Time.
Reliability, Availability and Maintainability

\[ \text{Avail} = \frac{MTBF}{MTBF + MTTR} \]
Software Reliability

Definition: the probability of failure-free operation of a computer program for a specified time in a specified environment.

Uses:
1. Understand user needs more precisely.
2. Evaluating and controlling quality level.

Requirement:

- Ultra reliable: $R > 1 \times 10^{-6}$
- Moderate reliable: $1 \times 10^{-3} \leq R \leq 1 \times 10^{-6}$
- Unreliable: $R < 1 \times 10^{-3}$
Software Reliability Modeling

- Fault Seeding Approach
- Time Domain Approach
- Data Domain Approach
- Coverage Domain Approach
- Coverage Enhanced Time Domain Approach
Time-Domain Approach

**Assumption:**  
- Times between failures are independent.
- Testing is representative of the operational usage.
- Time is used as a basis for the failure rate.

**Input data:** Time between failures or failure count.

**Estimate:** Mean time to failure (MTTF)
Operational Profile

- A profile is simply a set of disjoint alternatives with the probability that each will occur.

- An operational profile is a set of input states and their associated occurrence probabilities.
Time-Domain Approach

- Success
- Fail

Testing time

Failure intervals

\[ f_1 = t_1 \]
\[ f_2 = t_2 + t_3 \]
\[ f_3 = t_4 + t_5 + t_6 \]
\[ f_4 = t_7 + t_8 + t_9 + t_{10} \]
Problems With Time-Domain Approach

- Reliability estimate is affected by “saturation effect”.

- Effect of testing technique is ignored.

- Reliability estimate is sensitive to operational profile.
Saturation Effect

RESIDUAL FAULTS

Faults revealed

- $F_b$ (Functional)
- $F_{b\cup d\cup f\cup m}$
- $F_{b\cup d\cup f}$
- $F_{b\cup d}$

Testing effort ($t$)

Saturation region

Reliability

- $R_b$
- $R_d$
- $R_f$
- $R_m$

True reliability
Estimated reliability
Saturation Effect

Coverage statistics of \TeX and AWK

<table>
<thead>
<tr>
<th>Program</th>
<th>Coverages (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Block</td>
</tr>
<tr>
<td>\TeX</td>
<td>85</td>
</tr>
<tr>
<td>AWK</td>
<td>70</td>
</tr>
</tbody>
</table>
Reliability Estimation Tools

- SMERFS: Naval Surface Warfare Ctr.
- AT&T Toolkit: AT&T
- SRMP: City University, London
- CASRE: JPL & BellCore