Reliability Validation of Components (1)

Rev. July 19, 2006

Nobu Toge (KEK)

Introduction

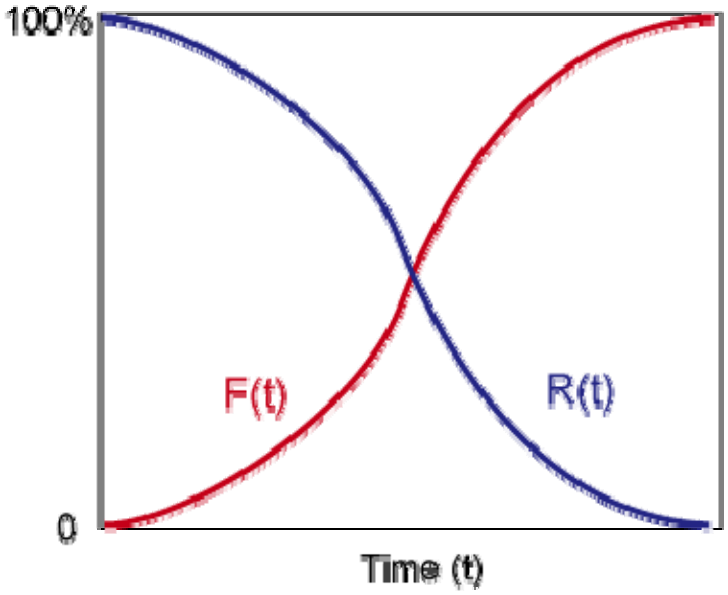
- First: Summary of the terminology and the types of failure profiles to consider.
- Second: Attempt at analyzing the level of reliability validation which might be possible at a test ML for ILC.
- Warning and disclaimer: I started serious reading of textbooks only last week → I can be VERY wrong.

References

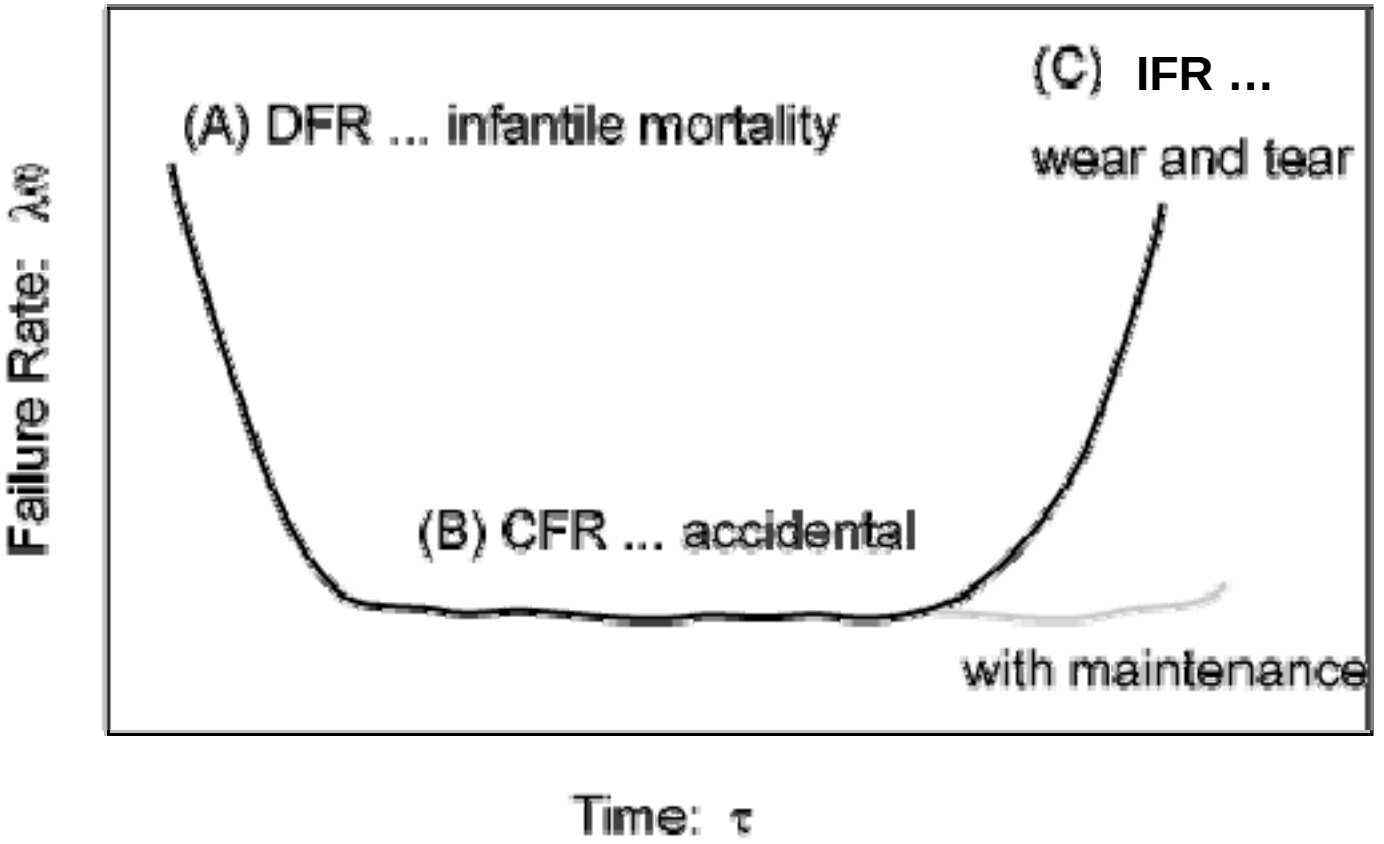
- Introduction to Reliability Engineering (信頼性工学入門） H.Shiomi　(塩見 弘）, Rev. 3, 2001, Maruzen (丸善）
- Reliability Engineering Series (日科技連信頼性工学シリーズ） 1984, Union of Japanese Scientists and Engineers (日科技連)
- Practical Reliability Engineering, P.O’Conner, 2002, John Willey and Sons. [Still waiting for delivery from Amazon]

Terminology

- Reliability Function: $R(t)$ = probability (or fraction) of items running without failure as function of time.
- Failure Distribution Function: $F(t)$ = probability of item failure as function of time. Note: $R(t) = 1 - F(t)$
- Failure Rate Function: $\lambda(t)$ = rate at which the items, who survived the preceding operation time of *t,* would fail: $\lambda(t) = -(d\,R(t)/dt) / R(t)$, hence $R(t) = \exp[-\int \lambda(s)\,ds]$
- MTBF = $\int R(t)\,dt$, where the integral is over $0 \rightarrow \infty$



Typical Time Profile of Item Failures (1)



Typical Time Profile of Item Failures (2)

- DFR (Decreasing failure rate distribution)
  - The $\lambda(t)$ is non-increasing func of t.
  - E.g. “initial” state (infantile mortality) where good and bad lots are mixed.
  - $R(t) = p \exp(-\lambda_1 t) + (1-p) \exp(-\lambda_2 t)$, with $\lambda_1 >> \lambda_1$
- CFR (Constant failure rate distribution)
  - The $\lambda(t)$ is ~constant.
  - E.g. “matured state” case where failures are random and accidental
  - $R(t) = \exp(-\lambda t) = \exp(-t/MTBF)$; $\lambda = 1/MTBF$
- IFR (Increasing failure rate distribution)
  - When $\lambda(t)$ is an increasing func of t.
  - Life limit due to wear and tear

Evaluation of MTBF (1)

- If a sufficient number (r > 15) of failures could be observed, an analysis which assumes a Gaussian distribution of TBF is likely to be adequate. i.e.,
- One can execute a standard “mean and sigma” analysis of failure times of the samples and compute the MTBF or estimate its upper/lower limits at adequate confidence levels.

Evaluation of MTBF (2)

- If only less than several instances of failure samples are available, the analysis may have to depend on the underlying model of $\lambda(t)$, which could be also unknown (catch-twenty-two situation).
  - Rescue formula: In case $\lambda(t)$ is assumed constant (CFR)
    - T = total operation time
    - r = # of failures observed in T
    - Then, $2r$ MTBF/<MTBF> will obey a $\chi 2$ distribution with DOF = $2r$

Evaluation of MTBF (3)

- If no failures are observed during the total operation time of *T* (either because *T* being too short or MTBF being too long), one can only estimate the limit value of MTBF or others. A couple approaches are possible:
  - Calculate the limit of reliability (which is usually not too useful anyways), or
  - Calculate the reliability and MTBF with a “worst case” assuming $r = 1$.
  - Calculate the limit of λ while assuming an exponential failure rate function.

Very Simple Case Study (1)

- 24 cryomodules (or whatever), each running over 1000 hrs, gave zero failure. What does this mean?
- This means zero failure in 24,000 total operation hours. OK. Still, what does this mean?
- Three types of analyses as per the previous page (only the results are shown. Consult textbooks for derivations):
  - Assuming Poisson distribution for # of failures (*r*), the lower limit (90% CL) of reliability over 24,000 hrs operation is ~0.9.
  - By taking the number of failure *r* = 1 as the most pessimistic scenario, we calculate the upper and lower limits (90% CL) of MTBF as:
    - $MTBF_U = 24x1000x19.4 = 4.6x10^5$ hrs, and
    - $MTBF_L = 24x1000x0.21 = 5040$ hrs
  - Assuming exponential distribution for the failure rate function with constant λ, the 90% CL of λ is given as $\lambda_u = 2.3/T_{total}$. Hence,
    - the $\lambda_u = 2.3/(1000x24) = 9.58x10^{-5}$.
    - $MTBF_L = 1/\lambda_u = 10{,}000$ hrs

Very Simple Case Study (2)

- We want to establish MTBF > 10^5 hrs with 90% CL for a kind of component. What should we do?
- We take the constant λ model. In case we try to evaluate $MTBF_L$ with T_{tot} hours of total operation time, in which zero failure is found:
  - $MTBF_L = 1/\lambda_u = T_{tot}/2.3$
  - $\rightarrow T_{tot} = 2.3 \times MTBF_L$ = 230,000 hrs is required.
- We need to observe zero failure with:
  - 192 units running in parallel for 1,200 hrs (50 days)
  - 24 units … for 9,600 hrs (400 days)
  - 8 units … for 28,800 hrs (1200 days)

Observations and Remarks for Further Study (1)

- Proper use of standard terminology is important. It is for discussing the reliability issues among parties with varying background and expertise. We should learn IEC 60050 (JIS Z 8115:2000) as the common language. Some teach-in might be worth, not only for S2/RDB but eventually for the entire GDE.

- Before discussing the issues with MTBF in the “constant failure” regime with confidence, we naturally have to address the issues with : “line debugging”, “infantile mortaility”and “initial burn-in”. We have to develop ways to separate these from the “constant failure rate” regime?

- A cursory look indicates that it will not be too easy to establish MTBF > 10^5 hrs with the level of test period and the number of units that are easily conceivable in pre-construction testing for ILC. Most likely these tests will only tell us if our production lines “are (or are not) contaminated by major bugs.”

Observations and Remarks for Further Study

(Continued) Therefore,

- Techniques of “accelerated testing” and “component-level mass testing” would be useful, but perhaps they are not applicable to all critical components.

- Techniques of FTA (Failure Tree Analysis) need to be looked into, also, and should be put into the perspective.

- Such efforts might go well beyond the original scope of S2, and could well be spelled out as the issue to address by GDE Engineering in the next N years.

- All I said here could be substantially wrong (since I am learning only recently). → Colleagues, please, cross-examine and check!