**Goel-Okumoto Software Reliability Growth Model**

[[](https://www.codingninjas.com/studio/profile/ead16948-444b-4f3a-be4d-4554e1b2960a)](https://www.codingninjas.com/studio/profile/ead16948-444b-4f3a-be4d-4554e1b2960a" \t "_blank)

[Author](https://www.codingninjas.com/studio/profile/ead16948-444b-4f3a-be4d-4554e1b2960a" \t "_blank)**[GAZAL ARORA](https://www.codingninjas.com/studio/profile/ead16948-444b-4f3a-be4d-4554e1b2960a" \t "_blank)**

**1 upvote**

**Share**

**Introduction**

A software reliability model's primary goal is to forecast the failure behavior of the software that will be experienced when the software is operating.

The estimation of remaining errors in the software is the deciding factor for the release of software or the amount of additional testing required. For the correct analysis of the remaining defects, software growth reliability models are applied.

One such model is The Goel-Okumoto model. Goel and Okumoto developed the Goel-Okumoto model (also known as the exponential NHPP model) in 1979. It is based on the following assumptions:

1. The number of errors discovered in each interval is independent of each other.
2. The quantities of available resources remain constant throughout the software's analysis.
3. When a failure happens, the fault that caused it is removed immediately without creating a new defect in the software.
4. The number of failures at time t follows a Poisson distribution with the mean value function μ(t). The boundary requirements for this mean value approach are μ(0) = 0 and Limit t→∞μ(t) = N < ∞.
5. The mean value function is designed in such a way that the expected number of error occurrences from time t to time t+Δt with Δt → 0 is proportionate to the expected number of undiscovered errors at time t.

**Goel-Okumoto Basic Model**

Because each fault is perfectly corrected after causing a failure, the number of inherent faults in the software at the start of testing equals the number of failures that will arise after an infinite length of testing.

By following assumptions 4 and 5, the following differential equation:

dμ(t)/dt = ∅[N - μ(t)]

where N is the estimated total number of errors in the software before testing and ∅ s the failure detection rate or fault intensity.

The mean value function solution to the above differential equation is provided by:

µ(t) = a (l-e-bt),        where a ≥ 0, b > 0

µ(t) = Predicted number of defects at time t.

a = Expected number of total defects in the code in infinite time (usually finite).

b = the rate at which the failure rate decreases = Roundness factor.

t  = Execution time/number of tests.

This model is called the Goel-Okumoto model.

In the case of the above-mentioned model, two parameters must be estimated: the total expected failures for infinite time (a) and the rate of reduction in the failure rate, also known as the roundness factor (b).

The parameters can be identified in two stages:

* during the testing.
* before the software is delivered to the client.

To estimate the parameters in terms of time, statistical inference methods such as Maximum Likelihood, Classical Least Squares, and Alternative Least Squares can be utilized.  
If the predictions are made after the product has been distributed to the client, they are made using software attributes such as the size and complexity of the software or failure data. Once the failure data in terms of execution time is known, these parameters can be calculated using any of the above-mentioned approaches.  
The parameters' accuracy generally increases with the size of the failure sample.  
  
Also see, [V Model in Software Engineering](https://www.codingninjas.com/studio/library/v-model-in-software-engineering) and [Introduction to Software Engineering](https://www.codingninjas.com/studio/library/software-engineering).

**FAQs**

1. **What is the reliability growth model in software engineering?**  
   A reliability growth model is a model of how system reliability changes over time during the testing process. When system failures are detected, the underlying defects that are generating these failures are addressed, and the system's reliability should improve via system testing and debugging.
2. **Who designed the GOEL-OKUMOTO Software Reliability Growth Model?**  
   Goel and Okumoto developed the Goel-Okumoto model in 1979. It is also known as the exponential NHPP model.

**Key takeaways**

In this article, we learned about the GOEL-OKUMOTO model in detail. Goel and Okumoto designed it in 1979. It is also known as the exponential NHPP model.

This mathematical equation of the model is given below.

µ(t) = a (l-e-bt),        where a ≥ 0, b > 0

µ(t) = Predicted number of defects at time t.

a = Expected number of total defects in the code in infinite time (usually finite).

b = the rate at which the failure rate decreases = Roundness factor.

t  = Execution time/number of tests.

The model provides a reasonable estimation of predicted errors.